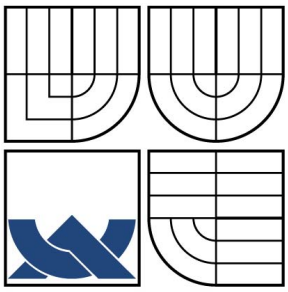
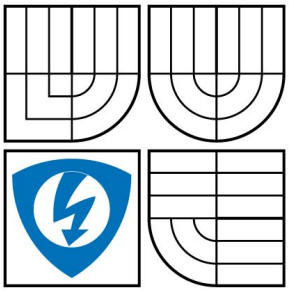


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

SPRÁVA SÍTÍ NA BÁZI PROTOKOLU IP

MANAGEMENT OF DATA NETWORKS BASED ON IP PROTOCOL

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

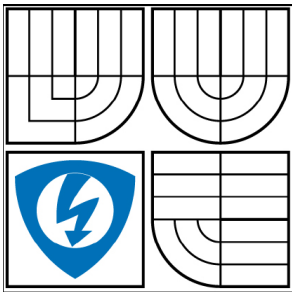
AUTOR PRÁCE
AUTHOR

Bc. PETR PATALA

VEDOUCÍ PRÁCE
SUPERVISOR

doc. Ing. VÍT NOVOTNÝ, Ph.D.

BRNO 2011



VYSOKÉ UČENÍ

TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav telekomunikací

Diplomová práce

Magisterský a navazující studijní obor
Telekomunikační a informační technika

Student: Petr Patala ***ID:*** 72985
Ročník: 2 ***Akademický rok:***2011/2012

NÁZEV TÉMATU:

Správa sítí na bázi protokolu IP

POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s problematikou dohledu nad funkčností a výkonností prvků datových sítí IP jako jsou přepínače a směrovače a spoje pomocí protokolu SNMP. Pomocí řešení SNMPc 7.1 od společnosti Castle Rock, případně i dalších volně dostupných ekvivalentů zjistěte možnosti získání informací z jednotlivých prvků dohledované experimentální sítě. U vybrané prvku uveďte důležité prvky databáze informačních prvků MIB. V experimentální síti realizujte různé typy provozu, vytvořte i různé chybové stavy, jako výpadek spoje, spojovacího prvku, smyčka v síti Ethernet, apod., a zjistěte, jak lze co nejsnáze problémy lokalizovat pomocí vyhodnocení dat z dohledového centra, případně s pomocí dalších dostupných síťových nástrojů. Na základě výsledků navrhněte doporučení pro úpravu sítě vedoucí ke zmírnění následků poruch či přetížení částí sítě. Navrhněte dvě laboratorní úlohy.

DOPORUČENÁ LITERATURA:

[1] Stephen J. Bigelow Mistrovství v počítačových sítích. Computer Press, ISBN: 80-251-0178-9, CR, 2004
[2] James M. Kretchmar, Libor Dostálek Administrace a diagnostika sítí. Computer Press, ISBN: 80-251-0345-5, 2005
[3] Castle Rock Dokumentace k produktu SNMPc Network Manager. Castle Rock, http://www.castlerock.com/products/snmpc/default.php, 2008

Termín zadání: 7.2.2011 ***Termín odevzdání:*** 26.5.2011

Vedoucí práce: doc. Ing. Vít Novotný, Ph.D.

prof. Ing. Kamil Vrba, CSc.
Předseda oborové rady

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

ABSTRAKT

Tato diplomová práce se věnuje problematice monitorování a správy počítačových sítí pomocí protokolu SNMP a jeho praktickém využití. Hlavní část práce popisuje práci s programem SNMPc v experimentální síti, implementací jeho částí do sítě, a také konfigurací SNMP agentů na směrovače, přepínač a koncové stanice. V práci jsou zahrnuty výsledky testování provozu, výpadku spoje, vliv zatížení prvků na parametry QoS, vytváření dlouhodobých statistik, prahů a alarmů. Práce uvádí také získané parametry protokolem SNMP ze síťových uzlů a koncových stanic.

KLÍČOVÁ SLOVA

SNMP, UDP protokol, síťová architektura, měření, monitorování, výstupní parametr, síťový prvek, agent, server, poller, směrovač

ABSTRACT

The objective of this master's thesis is monitoring and management of computer networks via SNMP protocol and its practical application. The main part describes working with SNMPc program in an experimental network through implementation of its parts into the network and configuration of SNMP agents on routers, switch and end station. This thesis includes the results of traffic testing, disconnected links, effects of traffic load on QoS parameters, making longterm statistics, baselines and alarms. The thesis also includes parametres obtained with SNMP protocol from network nodes and end station.

KEYWORDS

SNMP, UDP protokol, network architecture, telemetry, monitoring, output parameter, network element, agent, server, poller, router

Bibliografická citace práce:

PATALA, P. *Správa sítí na bázi protokolu IP.* Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2011. 88 s. Vedoucí diplomové práce doc. Ing. Vít Novotný, Ph.D..

Prohlášení

Prohlašuji, že diplomovou práci na téma Správa sítí na bázi protokolu IP jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne ....

podpis autora

Poděkování:
Děkuji vedoucímu diplomové práce doc. Ing. Vítu Novotnému, Ph.D. za velmi užitečnou metodickou pomoc a cenné rady při zpracování diplomové práce.

OBSAH

1 ÚVOD

Pro bezproblémový provoz jakékoliv počítačové sítě je potřeba, aby se o ni někdo staral. Obecně platí, že čím větší a komplikovanější síť, tím větší nároky na ni jsou kladeny a tím je také složitější její správa. Při administraci malé sítě, administrátor většinou nepotřebuje žádné výjimečné zařízení či software a dokáže ji spravovat ručně. Jestliže se objeví chyba, snadno ji lokalizuje a opraví. V moderní době však dosahují firemní sítě velikosti od několika stovek přes několik tisíců až po statisíce zařízení a jejich správa vyžaduje vysoce vzdělané odborníky a velmi specializované prostředky, které dokáží udržet síť v dobrém stavu. Vedení firem si uvědomuje, že investice vložené do správy sítě jsou daleko menší, než kdyby při jakékoli síťové poruše ztratila připojení například do veřejné sítě, nebo s částí společnosti. Často se můžeme setkat s celým oddělením nebo firmou, která se zabývá pouze správou sítě.

Aby bylo možné mít kvalitní celkový přehled o síti, je nutné sledovat její vytížení, zda nedochází k zahazování paketů, či velkému zpoždění, a mít možnost tyto informace zaznamenávat a zpracovávat. Je také nutné být včas informován, nastane-li v síti problém: přerušení kabelu, neplánovaný restart směrovače a podobně.

Tato práce se zabývá protokolem SNMP (Simple Network Management Protocol), který tyto požadavky umí zabezpečit, a umožňuje tak kvalitní správu a dohled nad počítačovou sítí. Jedná se o nejrozšířenější protokol v oblasti managementu sítě od roku 1989 a v současnosti existují již tři verze.

Diplomová práce je rozdělena do tří částí. První část popisuje základní informace o problematice moderní správy sítí z různých pohledů. Dále umožňuje čtenáři získat detailní informace o protokolu SNMP, jeho historii, verzích a způsobu přenášení informací a nakonec se souhrnem jednotlivých SNMP zpráv. Ty byly zasílány protokolem na experimentální síti v laboratoři, jejichž formát a obsah je názorně popsán při zachycení softwarem analyzujícím síťový provoz. O správu experimentální sítě se stará komerční program SNMPc.

Jeho co nejefektivnějším zavedením do experimentální sítě se zabývá navazující část práce. Jsou zde také rozepsány a testovány způsoby nastavení všech součástí dohledového softwaru SNMPc. Tato část práce se také věnuje implementaci protokolu SNMP na síťové prvky: směrovače, přepínače a koncové stanice.

Závěrečná a nejobsáhlejší část práce se věnuje testování a simulacím událostí v experimentální síti, které by se měly přiblížit podmínkám ve skutečné podnikové síti. K zaznamenání a vyhodnocení důležitých parametrů sítě jsou využity schopnosti programu SNMPc. Administrátor se tak může po správné konfiguraci tohoto programu rychle dozvědět požadované informace o síti. V této kapitole jsou simulovány výpadky spoje a reakce SNMPc, překročení prahových hodnot nastavených administrátorem a oznamování takových situací a další. Závěrem této kapitoly jsou uvedeny nejdůležitější hodnoty, které je možné získat ze správně nakonfigurovaných prvků podporujících protokol SNMP. Správce sítě tak může získat velmi podrobné informace z každé koncové stanice nebo síťového uzlu, a provádět tak optimalizaci síťových procesů.

Součástí diplomové práce jsou také dvě laboratorní úlohy obsažené v příloze. Úlohy by měly pomoci studentům osvětlit problematiku dohledu a správy sítě pomocí protokolu SNMP. Naučí se jak konfigurovat tento protokol na směrovači a také, jak co nejlépe využít funkce programu SNMPc k monitorování a správě sítě.

2 DOHLED A SPRÁVA POČÍTAČOVÝCH SÍTÍ

Požadavky kladené na datové sítě každým dnem rostou, ať už se jedná o množství uživatelů využívajících tyto sítě nebo vyžadovanou kvalitu, spolehlivost, dostupnost, cenu a jiné. Aby bylo možné tyto požadavky zabezpečit, je nutné vědět, co přesně se na síti děje. O tyto záležitosti se starají především správci sítě a počítačoví analytici. Tito profesionálové musí mít přehled nad hardwarem i softwarem sítě a jsou zodpovědní za údržbu, konfiguraci, aktualizaci, monitoring i zabezpečení aktivních síťových prvků, stejně tak jsou zodpovědni za udržování dobrého stavu spojů a rychlé reakce na výpadky a poruchy. Souhrnem tyto činnosti označujeme jako správa sítě, anglicky Network Management. V minulosti, kdy síť obsahovala jenom několik desítek uživatelů, souborový server a sdílenou tiskárnu, byla správa relativně jednoduchá. Dnes sítě obsahují tisíce uživatelů, desítky různých serverů, mnoho aktivních síťových prvků, jako jsou směrovače (router), přepínače (switch), rozbočovače (hub) a mosty (bridge). Každý server může běžet na jiném operačním systému a provozovat různé služby a každý aktivní prvek může být od jiného výrobce a nabízet různé funkce. Potom se ze správy sítě stává velmi náročná, o to však důležitější činnost. Oddělení správy sítě musí být připraveno zvládat všechny různorodé potřeby uživatelů na služby provozované v sítích a udržovat síť v takovém stavu, aby byla schopna vyhovět všem komunikačním potřebám. Hovoříme tedy o komplexní správě počítačových sítí.

Ze studií firem Forrester research nebo IDC-International Data Corporation vyplývá, že ve firmách zaměřených na informační technologie téměř polovina nákladů věnovaných síťovým technologiím jde na jejich management. Zbývající prostředky jdou na hardware, software a školení profesionálů.

Mnoho správců sítí řešilo i v současnosti řeší své problémy takzvanou „hasičskou technikou“. Znamená to, že neřeší problémy, které ještě nenastaly. Problém začnou řešit až v momentu, kdy se objeví , a teprve tehdy začnou pracovat na jeho vyřešení a na odstranění následků. Moderní správa sítí ale směřuje k trendu, kdy je nutné si vytýčit cíle o účelu, úrovní a kvalitě služeb, které chce firma nabízet. Je potřeba mít jasnou představu, jak těchto cílů dosáhnout a mít strategii, která je závislá na konkrétním řešení a možnostech firmy. I při zvýšení nákladů při takovémto plánování (a to se mnoha firmám nelíbí) je nutné si uvědomit, že optimalizace síťových prostředků a výkonu a komplexní dohled se v porovnání se ztrátami utrpěnými při havárii komunikačního systému a s financemi vynaloženými na odstranění jejích následků vyplatí. Jestliže se ještě před samotným zaváděním sítě správně rozhodneme a analyzujeme potencionální možnosti vzniku problémů, můžeme je vyřešit, ještě než nastanou, a tím těmto problémům předcházet, a tak ušetřit finance. Kvalitním plánováním můžeme zvýšit návratnost vynaložených investic a dosáhnout i maximální efektivity síťových technologií. Odhalení úzkých míst sítě nebo předvídání komplikací či nedostatečnou výkonnost síťových prvků v závislosti na zvýšeném provozu, vždy usnadňuje práci a je o mnoho jednodušší, než hledání a odstraňování chyb pak vzniklých.

Z hlediska velkých i malých firem a poskytovatelů síťových služeb je jejich garance velmi důležitá. Společnost IDC vypracovala pro firmu Novell studii , ve které poukazuje na úspory nákladů spojené s kvalitním managementem sítě. Jedná se o možnosti snížit počty zaměstnanců IT a přitom zvýšit kvalitu managementu a množství spravovaných síťových prvků, serverů a klientů. Administrátoři také nemusí trávit čas zbytečnou každodenní správou systémů a mohou se věnovat právě analýze a preventivnímu plánování. V neposlední řadě záleží především na spokojenosti koncových uživatelů, protože ty málokdy zajímá, proč síť nepracuje tak, jak očekávali. Síťový management také zmenšuje reakční dobu při výpadcích sítě, tj. odhalováním poruch a jejich řešení.

2.1 ARCHITEKTURA SÍŤOVÉHO MANAGEMENTU

Moderní architektury síťové správy jsou založeny ve většině případů na modelech manažer – agent, což je vlastně architektura klient – server (dotaz – odpověď),

- **Manažer**, nebo-li NMS (Network Management System) je správcem dohledového systému a komunikuje s agenty pomocí zpráv. Tato komunikace je založena na systému dotaz – odpověď. Manažer pošle agentovi dotaz, v němž žádá o určité informace, a následně čeká na odpověď. Agent odpoví MNS a zašle odpovídající nasbírané hodnoty. Tímto způsobem shromažduje server důležitá data o prvcích v síti např. počet a druh uzlů (aktivních prvků), verzi jejich používaného firmwaru, množství jejich vyrovnávacích pamětí, množství a druh přenášených dat, číslo běžícího procesu. Nasbírané informace vyhodnocuje, a zpracovává do podoby čitelné pro administrátora, jako tabulky či grafy. Z těchto dat je možné určit, co se přesně v sítí stalo, kdy, kde a jak sledovaná zařízení fungovala.

- **Agent** (klient) je software nainstalovaný na zařízení (Network Component), ze kterého plánujeme získávat data. Tento agent neustále monitoruje dané zařízení a sbírá i data o jeho stavech, chodu a funkcích, které vykonává. Všechna tato data jsou uložena v paměti v takzvané management information database (MIB). Agent zajišťuje komunikaci se serverem v podobě přijímání jeho dotazů a odesíláním odpovědí. Může také automaticky odesílat data v opakujícím se intervalu, nebo při určité předem definované situaci, či neočekávané chybě. Takováto zpráva se nazývá Trap. Zasílání těchto nevyžádaných zpráv agentem je velmi důležité při rychlém reagování například na výpadek spoje, nebo restart hardwaru. Kdyby měl agent čekat na další sekvenci sběru dat manažerem (tzv. polling), administrátor by se tak o chybě dozvěděl o až o několik sekund, či minut později.

- **Proxy agent** – aby nemusel být nainstalován agent na každém prvku sítě, je možné využívat tzv. proxy agenta. Může být nainstalován na síťovém prvku a podílet se na sběru informací do MIB (Management Information Base). Proxy agenta je možné využít na sběr informací do společné MIB zařízení odlehlé sítě, nebo na sběr údajů, které se příliš často nemění. Takovýto sběr informací se nazývá caching. Jedná se tedy o zmenšení požadavků NMS na procházení MIB. Je možné monitorovat i vzdálenou síť i v případě, že linka spojující proxy agenta a NMS je pomalá. Standardní SNMP provoz by spoj o mnoho více zatížil. Jako dokonalého proxy agenta můžeme brát například NCE (Network Control Engine). Tato zařízení jsou vestavěná jako moduly do výkonných modulárních koncentrátorů.

Je důležité zvolit potřebný kompromis pro stahování informací z agentů. Zbytečně časté vyžadování zaslání dat a neustálým monitorováním agentů by zatěžovalo síť, hlavně ve větších sítích. Také se může stát, že při správě velkého množství dat významně zatížíme správcovskou konzolu. Zasílání trapů tak umožňuje méně časté dotazování ze strany NMS, , .

2.2 MODEL SPRÁVY SÍTÍ PODLE ISO

Mezinárodní organizace pro standardizaci – ISO (International Organization for Standardization) vytvořila standardy i pro síťovou správu. Základních pět funkcí síťového managementu je definováno v ISO modelu, který je popsán v dokumentu označeném jako OSI Management Framework. Jedná se o část OSI basic Reference Model (ISO/IEC 7498-4) standardu, popisujícího síťový komunikační model.

- **Správa konfigurace** (configuration management) – tento dohled a monitorování sítě a její konfigurace se zaměřuje na zkoumání vlivu síťových prvků na chod síťových operací. Může se jednat o fyzické prvky – přepínače, rozbočovače, modemy, servery, hostitelské stanice, fyzickou vrstvu sítě – kabeláž, konektory atd. Také se může jednat o logické prvky – aplikace, protokoly, veškeré operační systémy a nastavení. Všechny nashromážděné konfigurační informace by se měly ukládat do databáze pro další použití.

- **Správa výkonu** (performance management) – jedná se měření zatížení a výkonnosti jednotlivých systémů sítě. Pokud známe dostatečné množství těchto parametrů (zatížení linek, odezvy aplikací, zaplnění front…), je možné reagovat dvěma způsoby:

  - ~ Reaktivní management – jestliže budeme kontinuálně monitorovat parametry sítě a při nastavení jistých prahových úrovní těchto parametrů, můžeme reagovat na jejich překročení. Tím vyvoláme určitou akci, například odeslání varovné zprávy.
  - ~ Proaktivní management – Pomocí různých plánovacích a simulačních metod je možné zjistit budoucí vlivy rozšiřování nebo jiných změn topologie, či výměny zařízení naší sítě na její chod. Je možné tak předem odhalit negativní dopad, který by tyto změny mohly mít - například na zatížení síťového prvku.

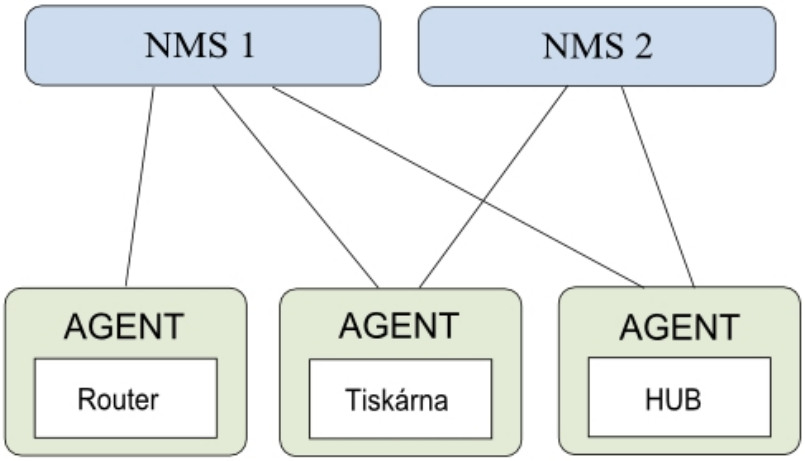
- **Účetní a evidenční správa** (accounting management) – toto monitorování sleduje využití sítě jednotlivými uživateli. S těmito informacemi je možné následně lépe plánovat růst sítě, využití jednotlivých prvků, změny sítě v závislosti na jejich využívání, či nevyužívání. Patří sem také regulování přístupu uživatelů k jednotlivým zdrojům a účtování za využívání poskytovaných služeb.

- **Správa bezpečnosti** (security management) - tento management se stará o řízení přístupu k síťovým prvkům pouze autorizovaným uživatelům, aby nemohlo dojít k jejich zneužití a neoprávněnému přístupu. Monitorují se také pouhé pokusy o přístupy do sítě a možnost o zakázání uživatelů, kteří nesplňují určité podmínky.

- **Správa poruch a chyb** (fault management) – správa detekce poruch sítě se také stará o jejich izolaci a zaznamenávání chybových hlášení do databáze. Systém může také zaslat upozorňující zprávu o události, nebo se dokonce pokusit chybu sám opravit.

2.3 SNMP (SIMPLE NETWORK MANAGEMENT PROTOCOL)

Protokol SNMP (Simple Network Management Protocol) je standardizovaný protokol aplikační vrstvy pro správu sítě a síťových zařízení, definovaný IETF (Internet Engineering Task Force) . S novým rozvojem a rozšiřováním počítačových sítí bylo nutné zavést prostředek, který by dovoloval vzdálenou správu síťových prvků a možnost jejich podrobnějšího nepřetržitého dohledu. Vývoj protokolu, který by splňoval takovýto účel, započal v roce 1988 jako reakce na nutnost sestavení dobře a jednoduše fungujícího systému pro správu počítačových sítí. V roce 1989 vychází první verze protokolu SNMPv1, která se zdá být pro tehdejší potřebu sítě dostačující. Je to jednoduchý aplikační protokol, částečně vycházející z protokolu SGMP (Simple Gateway Monitoring Protocol). SGMP byl vyvinut v roce 1987 pro vyměňování informací mezi směrovači. Další variantou pro správu sítí byl protokol CMIP (Common Management Information Protocol), který také vycházel ze SGMP, ale nedostal se takové podpory ze strany uživatelů ani vývojářů. Na rozdíl od SNMP byl SGMP objektově orientovaný a také pracoval s daty nad databází MIB (Management Information Base). Časem se však protokol SNMP ukázal jako nejuniverzálnější a nejdůmyslněji propracovaný. Dokázal pružněji reagovat na vzrůstající potřebu správy sítí a také díky jeho jednoduché implementaci a stal od počátku 90. let prakticky nejpoužívanějším protokolem pro správu sítí a řízení kvality služeb. V současnosti je nainstalován skoro na každém sofistikovanějším síťovém zařízení a jedná se prakticky o standard zajišťující správu konfigurace, kapacity, bezpečnost a celkovou správu sítí založených na protokolu TCP/IP (Transmission Control Protocol/ Internet Protocol), , .



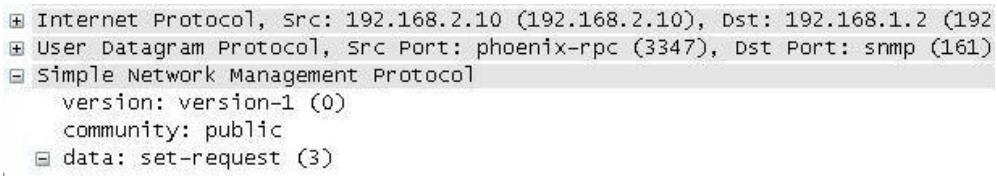
Obrázek 1: Architektura SNMP

SNMP je část internetové síťové řídicí architektury, využívá ho řada aktivních síťových prvků, jako jsou tiskárny, směrovače, přepínače, rozbočovače a další. Tento protokol mohou mít implementovány i pracovní stanice a servery jako nainstalovaný software, podobně jako protokol TCP nebo UDP, nebo může být přímo součástí operačního systému. Jedná se o asynchronní protokol na bázi klient-server, to znamená dvoubodovou komunikaci. Na straně klienta (zařízení, které chceme monitorovat) vystupujte agent, na straně serveru - NMS, který stahuje a zpracovává data a přijímá trapy zprávy.

Tento způsob komunikace má výhodu v tom, že zařízení s nainstalovaným agentem, není závislé pouze na jednom serveru, ale může odpovídat na žádosti několika na sobě nezávislých serverů. Stačí, když každý z nich zašle vlastní požadavek. Existuje řada příkazů a typu zpráv, kterými mohou komunikovat NMS a agenti mezi sebou, nebo NMS navzájem, tyto zprávy budou podrobněji popsány v dalších kapitolách.

Pro výměnu SNMP zpráv se vyžívá komunikace založena na protokolu UDP (User Datagram Protocol). UDP se skvěle hodí pro komunikaci klient server. Jedná se o nepotvrzovanou, nespojovanou komunikaci, kdy datagramy nezatěžují síť zbytečnou režií, potvrzováním a velkou hlavičkou, jako je tomu u protokolu TCP. Tato výhoda se násobí hlavně při velkém množství agentů a serverů, což systémy využívající SNMP většinou splňují. O kontrolu doručených zpráv se stará aplikační vrstva. V případě, že se zpráva na cestě ztratí, nebo poškodí, vyprší časovač na straně NMS a dotaz se opakuje.

Nevýhodou komunikace SNMP verze 1 a 2 je fakt, že přenos zprávy samotné probíhá nezabezpečeně. Paket obsahuje pouze textové řetězce a v případě, že je zachycen, lze z něj tento řetězec snadno přečíst - jak je možné vidět na zachycené zprávě pomocí programu Wireshark (viz Obrázek 2). Zkušený programátor tak může převzít částečnou kontrolu nad zařízením, aniž by měl oprávnění. Tento problém však nezpůsobuje protokol SNMP, ale jeho implementace na agentovi, případně manažerovi.



Obrázek 2: SNMP community string

SNMP agenti naslouchají na portu 161, manažeři se napojují na dynamický port, který si zvolí, aby mohli současně bez problému komunikovat s různými agenty. Agent pak odpovědi posílá na port, odkud mu přišel dotaz. Při nestandardní situaci, jako poškození zařízení apod., by ale agent nevěděl, kam má poslat zprávu TRAP, proto byl pro tyto situace vyhrazen port 162. Realizace tohoto protokolu nemůže přijmout zprávu, která je delší než 484 oktetů.

2.3.1 Historie protokolu SNMP

SNMPv1 - První verze protokolu umožňovala správcům sítí poměrně značné možnosti oproti dosavadním řešením. Bylo však jasné, že protokol bude muset prodělat ještě řadu změn, aby se jeho využitelnost a flexibilita uplatnily v plné míře. SNMP operuje přes protokoly UDP, IP (Internet Protocol), CLNS (OSI Connectionless Network Service), DDP (AppleTalk Datagram-Delivery Protocol), a IPX (Novell Internet Packet Exchange). Protokol umožňuje snadnou implementaci agentů a jejich rozšiřitelnost v rámci rozrůstající se sítě.

V první verzi definované v RFC umožňovalo komunikaci a dotazování pět základních definovaných operací, které byly zapouzdřeny v datové jednotce PDU (Protocol Data Unit). Jednalo se o operace: GetRequest, GetNextRequest, GetResponse, SetRequest a Trap (blíže viz 2.3.4). Parametry přenášených řídících informací byly definovány v RFC 1155 a později v RFC 2578, .

Hlavní nevýhodou SNMPv1 bylo jeho nízké zabezpečení. Systém autorizace přístupu uživatelů byl na velmi nízké úrovni, hesla byla přenášena společně s daty nezabezpečenou cestou a bylo tak možné je zachytit. Další důležitou chybějící součástí bylo potvrzování SNMP zpráv v aplikační vrstvě, protokol UDP na transportní vrstvě to nezajišťoval.

V neposlední řadě chyběla možnost komunikace mezi manažery navzájem. To znamenalo, že informace nebylo možné sdílet mezi více NMS. Pokud se správce rozhodl mít více serverů pro správu jednoho systému, musel každý z nich samostatně pollovat data a to zbytečně zahlcovalo síť a navíc neměly servery přesně tatáž data.

SNMPv2 - Jako reakce na tyto nedostatky vyšla v roce 1993 druhá verze protokolu. Byla definována v RFC1441- RFC1452 a měla vyřešit stávající problémy. V SNMPv2 se uplatnily zkušenosti z předchozí verze a bylo přihlédnuto k potřebám rychle se rozvíjejícího telekomunikačního průmyslu. Byly přidané operace Get-Bulk, InformRequest, Report, Notification (blíže viz 2.3.4), které pomohly zvýšit efektivitu protokolu.

Kladen byl důraz také na zabezpečení, řešení nebylo ještě přivedeno k dokonalosti a zdálo se být příliš složité. Do druhé verze protokolu byla zahrnuta funkce potvrzování na aplikační vrstvě, takže stanice již mohly rozpoznat, zda byla nebo nebyla zpráva doručena. Také bylo uvedeno do provozu několik dalších verzí, a to komprimovaná SNMPv2c (známá také jako verze 1.5), která ještě neobsahovala zabezpečovací modul a používala systém verze 1. Další pokus byla verze SNMPv2u, která byla vlastně pokus o kompromis s cílem vytvořit lepší zabezpečení než ve verzi 1, ale neučinit systém příliš složitý. Verze byla uvedena na trh jako SNMPv2*; tento systém byl částečně použit i ve verzi nynější.

Velká nevýhoda SNMPv2 byla v tom, že nebyla vzájemně kompatibilní s SNMPv1. A to hlavně ve dvou základních věcech, v protokolárních operacích a ve struktuře zprávy, v hlavičce i PDU. Vzájemná konverze je definována v RFC1452 a je uskutečněna pomocí Proxy agenta a dvojjazyčného síťového řídicího systému (Bilingual Network-Management System).

SNMPv3 – V současnosti nejnovější verze protokolu SNMP, uvedená do provozu v roce 2004, IETF definovaná v RFC 3411-3418. V současnosti většina zařízení prakticky podporuje všechny tři verze protokolu. Asi nejvýznamnější změnou oproti starším verzím je nový přístup k zajištění bezpečnosti. Jestliže existuje požadavek ze strany aplikace na zabezpečení, protokol ho aplikuje ve dvou různých stupních:

- přenos zprávy,
- obsah zprávy.

Zabezpečení v přenosu zprávy je aplikováno jako Access Control (autentizace-ověření pomocí jména a hesla) a je začleněno do protokolových operací. Tento autorizační zabezpečovací model ustanovuje, který přístup řídicího objektu by měl být povolen. Tento model také definuje MIB modul užívaný během zpracování dat a metody vzdálené konfigurace přístupu. Komunikační zpráva používající autentizační protokol obsahuje pole autentizačních parametrů jako část zabezpečovacích parametrů. Toto pole je tvořeno oktetovým řetězcem, který je reprezentován prvními 12 oktety HMAC-SHA-96, což je autentizační modul poskytující služby pro vstup a výstup. Tato verze využívá výhod předchozích verzí a jeví se jako dostatečná pro potřeby moderních komunikačních sítí.

2.3.2 Formát zprávy SNMP

Aby bylo možné SNMP informace na jedné straně poslat a na druhé opětovně rozeznat a sestavit, bylo nutné vytvořit podobu datové jednotky PDU. Základní myšlenka byla, že každá vytvořená zpráva musí byt přiřazena určitému konkrétnímu objektu, který je popsán pomocí tzv. OID (Object Identifier). Takovýto jednoznačný identifikátor má dvě podoby. Pro člověka přijatelnější textová forma, ale pro snadnější počítačové zpracování zase numerická hodnota. Aby bylo možné lépe zacházet s objekty, byly seřazeny do hierarchické stromové struktury.

Parametr OID jednoznačně identifikuje jak objekt, tak i jeho umístění v této struktuře. Identifikátor se skládá z čísel oddělených tečkou, kde každé číslo označuje větev stromu například např.: 1.3.6.1.4.1.23., jeho textové vyjádření v tomto případě bude iso.org.dod.internet.private.enterprises.novell., řetězec je ukončen tečkou, za kterou je možné vytvářet další rozvětvení. Jednotlivé větve stromu byly definovány v RFC1066.

Typ a syntaxe spravovaných objektů je definovaný jazykem ASN.1 (Abstract Syntax Notation One). Tato syntaxe definuje přesně, jakou podobu mají přenášené informace nejen v protokolu SNMP, ale i jakémkoli jiném systému. Je to formální jazyk pro definici struktury dat v našem případě v komunikačním protokolu. ASN.1 definuje základní datové typy a také nové datové typy vzniklé kombinací typů základních, vhodných například pro definici tabulek a řádků obsahujících různé datové typy.

Mezi definované typy základní patří:

- **Integer** – Jednoduché celé číslo. Většina implementací podporuje rozsah do 32 bitů, ale přesná konečná hodnota není definována.
- **Counter** – Jedná se o nezáporné celé číslo, které se zvětšuje o konstantní přírůstek v rozmezí 0 až 2^{32}-1. Tento typ se využívá především pro účely počítání. Jestliže hodnota dosáhne maxima, začne opět cyklicky od 0.
- **TimeTicks** – Nezáporné celé číslo odpovídající času měřenému v sekundách od určité události. Hodnoty nabývají modulo(2^{32}-1). Využívá se například pro měření doby, kdy je v provozu nějaké zařízení. Je důležitá velikost - absolutní hodnota, nikoliv rozdíl dvou hodnot, jako je tomu u typu Counter.
- **Gauge** – Nezáporný integer, jehož hodnota může růst i klesat v určitém rozmezí a nikdy nemůže překročit maximální hodnotu
- **IP Address** – 32 bitová IP adresa
- **Object Identifier** – reprezentuje název uzlu
- **Octet String** – sekvence bajtů, která se používá k vyjádření řetězce znaků, nebo binárních dat.

Standard SNMP dovoluje tyto objekty strukturovat a ukládat do tabulek databáze MIB. Hodnoty takovéto tabulky jsou pak skalární hodnoty. Tabulky nelze vnořovat jednu do druhé.

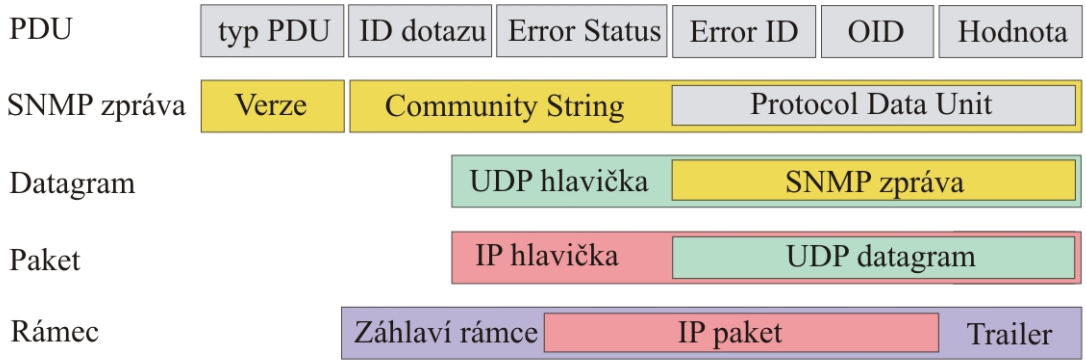
Vlastní formát SNMP zprávy je rozdělen na dvě části - hlavička paketu a protokolární datová jednotka zvaná PDU (Protocol Data Unit.) V hlavičce je uložena verze SNMP a community string pro zabezpečení. Jedná se o kombinaci jména a hesla, která funguje jako autentizace přístupových práv k agentovi. K jednomu agentovi totiž mohou přistupovat různí manageři s různými právy přístupu (Read Only / Read-Write). Tento řetězec je proto uložen v každém paketu, aby bylo možné komunikaci jednotlivých manažerů od sebe rozlišit.

Datová jednotka paketu obsahuje vlastní informace závisející na typu operace. Operace Trap má od ostatních operací (GetRequest, SetRequest…) odlišnou strukturu PDU. Jak ukazuje Obrázek 3, po hodnotě community string následuje samotné PDU, které obsahuje hodnoty:

- Typ PDU – určuje, zda se jedná o zprávu typu get, trap či jinou operaci,
- ID dotazu – označuje příslušné dvojice požadavků a odpovědí,
- Error Status – udává, zda byl požadavek úspěšný, nebo udává typ chyby, která nastala,
- Error ID – obsahuje podrobnější informace o chybě a přiřazuje jí určitou hodnotu,
- OID – identifikátor objektu,

- Hodnota – konkrétní hodnota proměnné.

Dvojice OID a hodnota představuje uživatelská data, a konkrétní hodnoty, které představují konkrétní veličiny. Manažer může v jednom dotazu požadovat několik různých informací a každá z nich bude mít vlastní dvojici OID-hodnota. Tyto dvojice jsou umístěny v poli proměnných (Variable Bindings), .



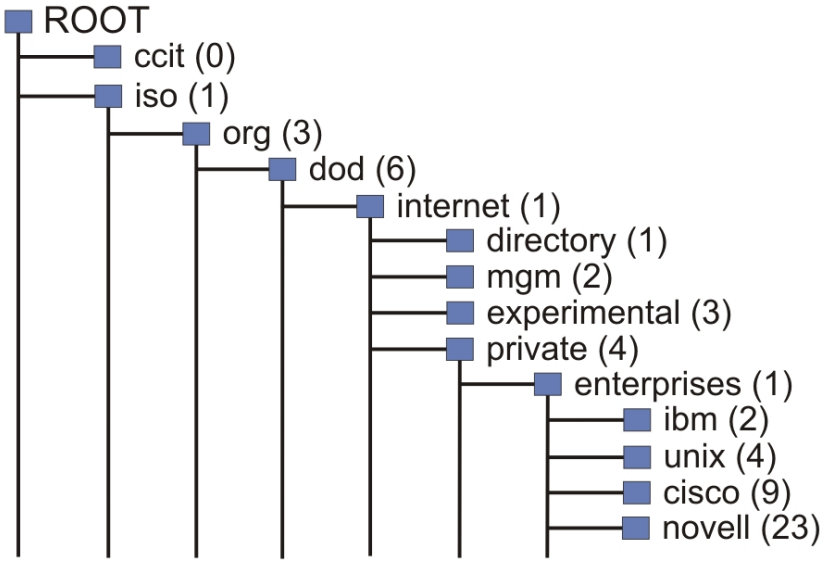
Obrázek 3: Formát zprávy SNMP

Zpráva Trap obsahuje ve svém PDU hodnoty:

- Enterprise – identifikace objektu, který zprávu vygeneroval,
- Agent address – obsahuje adresu objektu, který zprávu vygeneroval,
- Generic trap type – identifikace typu zprávy,
- Specific trap code – kód zprávy,
- Time stamp – časová značka udávající dobu mezi reinicializací sítě a vygenerováním zprávy,
- OID – identifikátor objektu,
- Hodnota – konkrétní hodnota proměnné.

2.3.3 SNMP a MIB databáze

Protokol SNMP sám nedefinuje, jak a které informace a proměnné by měl nabízet, ale využívá externí databázi MIB (Management Information Base). Specifikace databáze definuje jak strukturu, tak i formát dat (viz Obrázek 4). Do databáze se ukládají na základě definovaných instrukcí data předávána v SNMP. MIB je hierarchicky organizovaná databáze, která odpovídá všem SNMP zařízením a je objektově orientována jako sada SNMP objektů, operací a relací na objektech i mezi objekty. Kořen struktury (root) není pojmenován. Každá větev tohoto stromu je označena textovým řetězcem a odpovídajícím číselným identifikátorem OID. Stromový systém databáze MIB umožňuje křížení všech vrstev OSI modelu, vstupujících do aplikací, jako jsou emaily a databáze -RFC 1066.



Obrázek 4: Struktura databáze MIB

Každé SNMP zařízení může implementovat více než jednu MIB, a to podle funkce, kterou zastává. SNMP operace je nutné provádět nad jednotlivými objekty v tabulce a není možně operovat s celými tabulkami. Identifikace jednotlivých polí se provádí pomocí indexů. K objektům je možné přistupovat přímo pomocí operace Get a uvedení přesné pozice, nebo pomocí GetNext (viz 2.3.4), kdy se sekvenčně prochází všechny hodnoty tabulky.

2.3.4 SNMP operace

V první verzi protokolu SNMP je definováno těchto pět základních operací :

- Get
- GetNext
- GetResponse
- Set
- Trap

Ve verzích SNMPv2 a SNMPv3 jsou podporovány další operace, které reagují na nedostatky verze první :

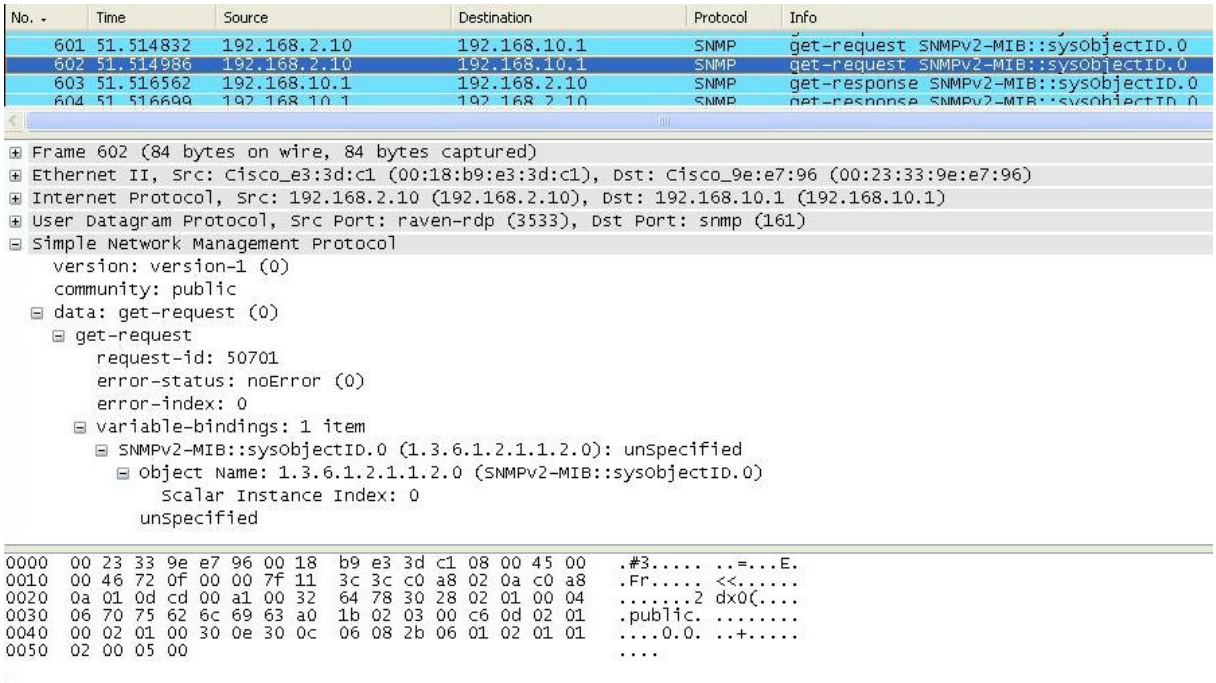
- GetBulk
- Notification
- Inform
- Report

Get

Tato operace je jednou ze základních operací protokolu SNMP. Zprávu obsahující tuto operaci generuje manažer v případě, že chce od určitého agenta získat informace. Informací se rozumí hodnota instance objektu obsažené v MIB tabulce agenta. Manažer nemusí pro každý požadavek od jednoho agenta generovat novou zprávu, ale umístí několik dotazů naráz do pole variable bindings (viz 2.3.2). Jelikož je možné, že existuje několik instancí určitého objektu (např. tabulka uchovávající informaci o přijatých a odeslaných paketech), užívá se jako odkaz na konkrétní instanci objektu OID, který přesně určí umístění v hierarchické struktuře i daný objekt.

Jakmile agent vyhledá požadovanou informaci, zašle manažerovi tato data pomocí zprávy GetResponse. Odpověď je identifikována stejným OID a obsahuje také hodnotu proměnné. Jestliže není možné zaslat manažerovi odpovědi na všechny dotazy, pak odešle pouze zprávu s patřičně vyplněnými poli Error Status a Error ID.

Definici operace Get podle – RFC 1067 si lze ověřit například užitím nástrojů pro zachytávání síťové komunikace. Pomocí paketového analyzátoru Wireshark je možné zachytit komunikaci zařízení s nainstalovaným agentem a MNS.



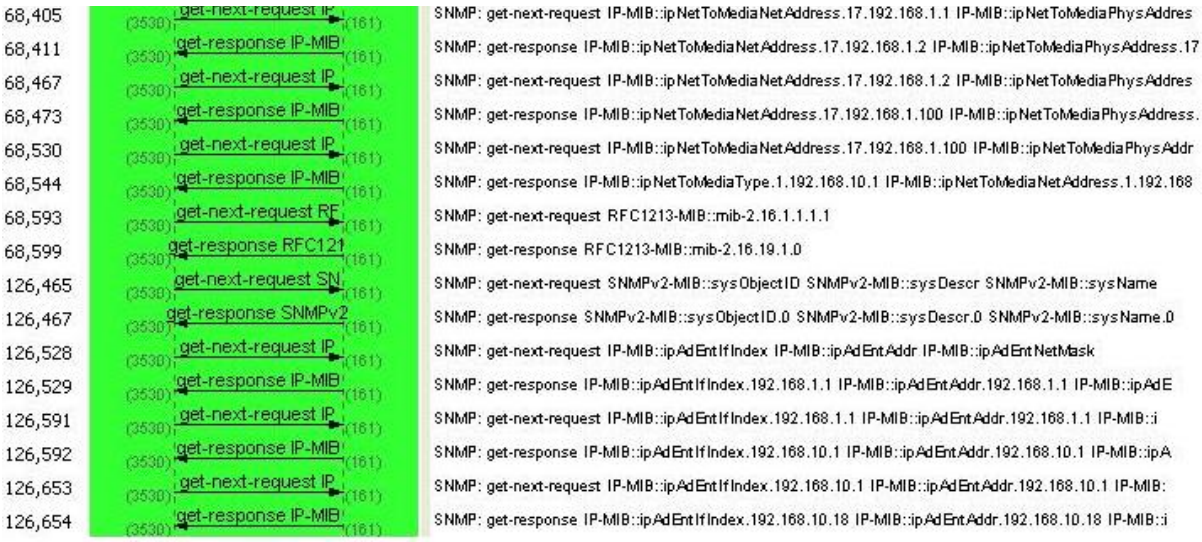
Obrázek 5: Struktura zprávy GetRequest

GetNext

Jedná se o podobnou operaci jako Get, s tím rozdílem, že agent vrací manažerovi vždy následující hodnotu. Manažer používá operaci Get v případě, že chce získat jednu nebo několik hodnot, avšak předem ví, kolik jich bude. Jedná-li se o velkou sadu hodnot, nebo není-li jejich počet manažerovi předem známý, zasílá manažer požadavek zprávou GetNextRequest (operace GetNext). Zpráva má podobný charakter jako jako GetRequest, ale adresace objektu je odlišná. Systém odkazuje na hodnotu v tabulce MIB a znovu zasíláním operace se index posunuje sekvenčně v tabulce a odesílá vždy následující hodnoty, dokud

nenarazí na konec. Poté signalizuje manažerovi odesláním chybové hlášky. Agentovou opovědí na dotaz je operace GetResponse stejně jako v předešlém případě.

Pro lepší přehlednost je možné se podívat na datový přenos mezi agentem a manažerem (viz Obrázek 6). V tomto případě chtěl manažer s IP adresou 192.168.2.10 získat velké množství dat, když nastala doba pro další polling dat na síti. Odeslal proto sadu dotazů GetNextRequest na port 161 různých agentů, např. na rozhraní směrovače 192.168.10.1 nebo přepínače 192.168.1.2. Manažer se dotazoval na síťové a fyzické adresy jednotlivých portů. Tito agenti mu odpovídali klasickou zprávou GetResponse na port, ze kterého byl dotaz odeslán.



Obrázek 6: Datový tok SNMP

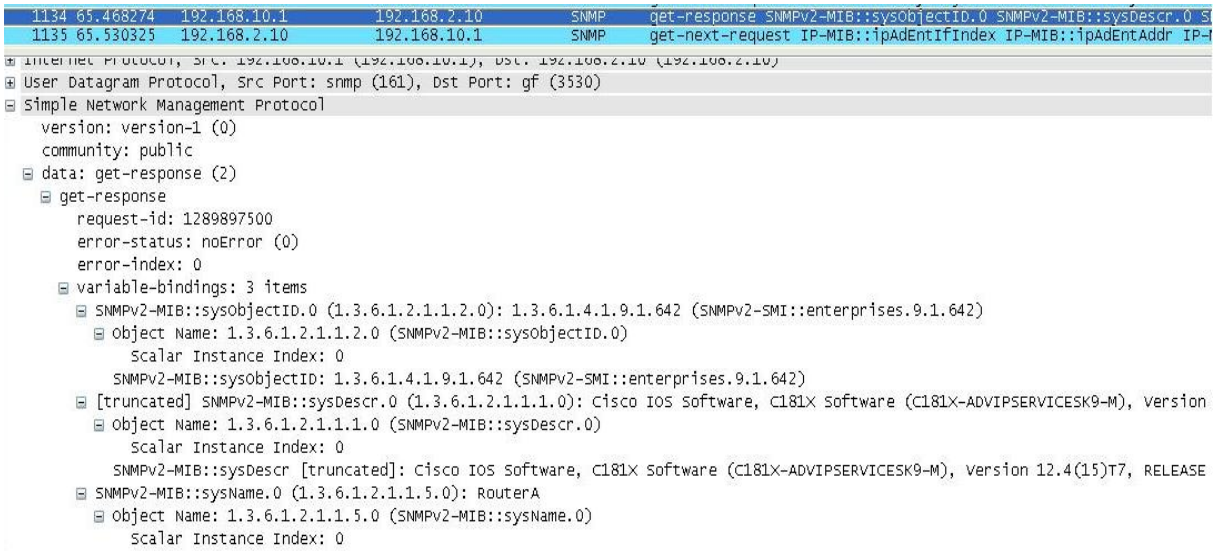
GetResponse

Zpráva tohoto typu je odpovědí na dotazy GetRequest (operace Get) a GetNextRequest (operace GetNext). Na oba typy dotazu je formát zprávy stejný. Odpovědí jsou buď hodnoty v poli variable bindings (variable binding list pro více hodnot zasílaných v jedné zprávě). Formát zprávy GetResponse je podobný jako zprávy GetRequest s výjimkou pole typ PDU a případných vyplněných chybových hlášek.

Jak je možné vidět ve struktuře zprávy (viz Obrázek 7), pole variable bindings může obsahovat několik OID (viz 2.3.2). Zde se jedná o odpověď na dotaz GetNextRequest z MNS s IP adresou 192.168.2.10 na směrovač s rozhraním 192.168.10.1. Tato odpověď zahrnuje:

- sysobjectID – Poloha tohoto objektu ve stromu MIB tabulky (OID) je 1.3.6.1.2.1.1.2 a jeho datový typ je Object Identifier. Jedná se o jednoznačný identifikátor prodejce subsystému síťové správy v dané entitě. Tato hodnota je přidělena do SMI enterprise větve (1.3.6.1.4.1) a poskytuje jednoduchý a jednoznačný způsob pro rozlišení o jaký druh zařízení je spravován.
- sysDescr – Objekt s OID 1.3.6.1.2.1.1.1 má datový typ DisplayString. Jedná se o datový typ 255 znaků dlouhého řetězce, ve kterém některé kombinace znaků mají grafický význam. Např.: CR LF znamená nový řádek. Objekt sysDesr má status pouze pro čtení a jedná se o textový popis zařízení, který by měl obsahovat název entity, typ používaného hardwaru, verzi operačního systému a síťového softwaru.

- sysName – OID objektu je 1.3.6.1.2.1.1.5 a jedná se o název zařízení přidělený administrátorem. V tomto konkrétním případě (viz Obrázek 7) je název RouterA. Tento objekt má oproti předešlým status Read-Write a je možné ji měnit uživatelem. SysName je opět datového typu DisplayString.



Obrázek 7: Struktura zprávy GetResponse

Můžeme si zde uvést další neméně důležité objekty náležící do větve iso(1).org(3).dod(6).internet(1).mgmt(2).mib-2(1).system(1).

- sysContact – Pozměnitelný objekt datového typu DisplayString slouží pro textové označení kontaktní osoby pro správu daného uzlu spolu se správou jak tuto osobu kontaktovat. V případě, že není text vyplněn, je délka řetězce nulová.
- sysLocation – jak už název napovídá, jedná se o označení lokace zařízení.
- sysServices – Hodnota tohoto objektu označuje nastavení služeb, jaké může zařízení teoreticky nabízet. Jedná se o součet, který nám ukáže na jakých vrstvách modelu OSI zařízení pracuje. Jedná-li se například o směrovač pracující na 3. vrstvě modelu. Vypočítá se hodnota jako $2^{(L-1)} = 2^{(3-1)} = 4$
- sysORLastChange – jedná se o objekt typu TimeStamp, který udává dobu, kdy byla na zařízení provedena poslední změna.
- sysORTable – uvádí možnosti lokální SNMP aplikace s ohledem na různé MIB tabulky. Dynamicky konfigurovatelné SNMP entity podporující MIB moduly, budou mít dynamické číslo rádků.

Může se také stát, že se při plnění požadavků manažera na agentovi vyskytnou určité problémy. V takovém případě ve zprávě GetResponse se vyplní pole Error Status a Error ID. Tyto chyby jsou předem definované a přenáší se pouze jejich textové označení. Jejich rozlišení vypadá takto

- noError(0) – agent nezaznamenal žádnou chybu během přenosu
- tooBig(1) – agent nemohl umístit požadovanou hodnotu nebo hodnoty do jediné zprávy

- noSuchName(2) – operace Get požaduje hodnotu která není agentovi známa
- badValue(3) – operace Set chce změnit určitou hodnotu, ale špatným hodnotou proměnné
- readOnly(4) – manažer požaduje změnu hodnoty na agentovi, ale nemá potřebná oprávnění podle community, nebo se pokouší změnit hodnotu, která má status Read Only
- genErr(5) – takto je označený jiný typ chyby, který není předem definován

SetRequest

Tato operace nabízí oproti předchozím operacím manažerovi možnost, aby také mohl měnit nastavení jednotlivých zařízení, a ne z nich pouze číst data. Manažer je tak schopen pomocí zprávy SetRequest změnit některé hodnoty v agentově MIB tabulce, nebo také přidat celý řádek. Je možné tyto hodnoty měnit hromadně, ale je nutné k nim přistupovat přesně. Stačí, aby se vyskytla jedna chyba, a celá operace je zrušena. Může se například stát, že manažer bude chtít změnit hodnotu, která má státu pouze pro čtení a v tom případě bude zaslána zpráva GetResponse s patřičným oznámením o chybě. Tato operace se často využívá v případech, kdy administrátor potřebuje například vzdáleně restartovat zařízení.

Trap

Jestliže nastane v systému chyba, nebo zvláštní situace, kterou je potřeba oznámit manažerovi, musel by se svým varováním agent vyčkat, dokud ho manažer nepožádá po uplynutí dotazovací periody. Jelikož protokol SNMP generuje značný provoz na síti, tak se při velkém počtu stanic může tato perioda pohybovat i v řádu několika minut. Reakční doba při detekci a odstraňování chyb by ale byla příliš pomalá. Proto byla definována operace Trap, ta umožňuje generovat agentovi zprávu informující manažera na nastalou situaci. Jedná se například o výpadek spoje, či uzlu v případě zahlcení sítě a podobně.

Je nadefinováno několik typů této operace několik typů této operace (RFC1157), jež jsou přenášeny jako numerické hodnoty v poli Generic trap type (viz 2.3.2):

- **The coldStart Trap (1)** – agent nebo protokolový prvek, sám sebe restartoval a nakonfiguroval, nebo pozměnil.

- **The warmStart Trap (2)** – agent nebo protokolový prvek, sám sebe restartoval, ale nenakonfiguroval, ani nepozměnil.

- **The linkDown Trap (3)** – agent nebo protokolový prvek rozpoznal selhání v jednom z komunikačních spojení, nakonfigurovaných agentem. Ve variable-bindings jsou obsaženy jméno a hodnota indexu instance postiženého prostředí.

- **The linkUp Trap (4)** – agent rozpoznal nové komunikační spojení. V poli variable-bindings jsou opět uvedeny důležité informace.

- **The authenticationFailture Trap (5)** – znamená, že adresát přijaté protokolové zprávy nemá řádné pověření. Zatímco realizace SNMP musí být schopna generovat tuto zprávu, musí být také schopna potlačit vyslání stejného Trapu adresátovi, pošle ho pouze na port 162.

- **The egpNeighborLoss Trap (6)** – agent rozpoznal že jeho soused (vrstevník) byl zrušen a jejich sousedství dále netrvá. V poli hodnot pošle jméno a hodnotu

zasaženého souseda. Egp je skupina v tabulce MIB (kapitola 2.3.3) poskytující informace součástech a stavu při používání protokolu EGP (Exterior Gateway Protocol).

- **The enterpriseSpecific Trap (7)** – agent zjistil, že se vyskytla určitá specifickou událost, tu identifikuje a odešle manažerovi.

GetBulk

Zpráva GetBulkRequest je generována a přenášena jako požadavek SNMPv2 aplikací a je spolu s dalšími zprávami definována v RFC 1448. Tato zpráva umožňuje dotázat agenta na velké množství dat, které pravděpodobně překračuje maximální definovanou velikost SNMP zprávy, ale zároveň se vyvarovat chybové hlášce tooBig. Zpráva jednoduše a rychle získá více hodnot z MIB tabulky, protože efektivně využívá zbývající místo ve zprávě. Odpověď tedy obsahuje maximální možný počet hodnot, který se do zprávy vleze. Jedna zpráva tak nemusí obsahovat všechny, ale je odeslána. Proces k vybrání hodnot, které budou odeslány, porovnává hodnoty polí non-repeaters a max-repetitions. Každou hodnotu variable binding zpracovává v závislosti na těchto dvou polích a umisťuje ji do zprávy GetResponse. Naopak v případě zpráv GetRequest, GetNextRequest a SetRequest, se zpracovává každá hodnota samostatně.

Inform

Zpráva nesoucí tuto operace je generována a odeslána manažerem k jinému manažerovi za účelem výměny informací uložených v jejich MIB. V první verzi SNMP komunikace mezi manažery nebyla vůbec podporována (viz 2.3.1). Cílové stanice, na které jsou zprávy Inform zasílány jsou uloženy v tabulce snmpEventNotifyTable.

Report

Ve druhé verzi protokolu SNMP byla definována také zpráva Report, která ale nebyla využívána. Jejím smyslem bylo umožnit komunikaci mezi SNMP agenty v případě, že dojde k problémům ve zpracování zpráv.

3 SNMP NA EXPERIMENTÁLNÍ SÍTI

3.1 PROGRAM SNMPC V7.103

Tento produkt vyvinula a představila trhu společnosti Castle Rock Computing. Jedná se o první software umožňující komplexní správu sítí pro operační systém Windows™. Program využívá protokol SNMP pro monitorování a správu sítě ve všech jeho verzích SNMPv1, SNMPv2c a SNMPv3. Na trhu se již objevila nová verze SNMPc 7.2. Program SNMPc je nabízen ve dvou základních balíčcích :

- **Enterprise** – na server (NMS) může být připojeno naráz několik uživatelů. První z nich přímo na serveru a další na vzdálených konzolách. Hlavní server je zálohován vedlejším (backup) serverem, který při výpadku zaujme pozici hlavního a kompletně ho nahradí. Enterprise také umožňuje pokročilé reportování stavu sítě pomocí webových stránek. Lze rozšířit o SNMP Remote Access Extension, přes který je možno připojit neomezené množství konzol a vzdálených polling agentů. Rozšiřující balík také, na rozdíl od samotné enterprise edice, podporuje JAVA konzole

- **Workgroups** – jedná o jednodušší verzi Enterprise, kde je možno připojit pouze jednoho správce, který může monitorovat maximálně 1000 objektů (ideální pro LAN-MAN sítě), nepodporuje možnost pokročilého reportování

- **SNMPc Online 2009** – jedná se nový zásuvný modul pro SNMPc Enterprise edici, který umožňuje ukládat a spravovat celou síť online. Obsahuje také možnosti pokročilého reportování a přístup ke všem datům na síti přes webový prohlížeč.

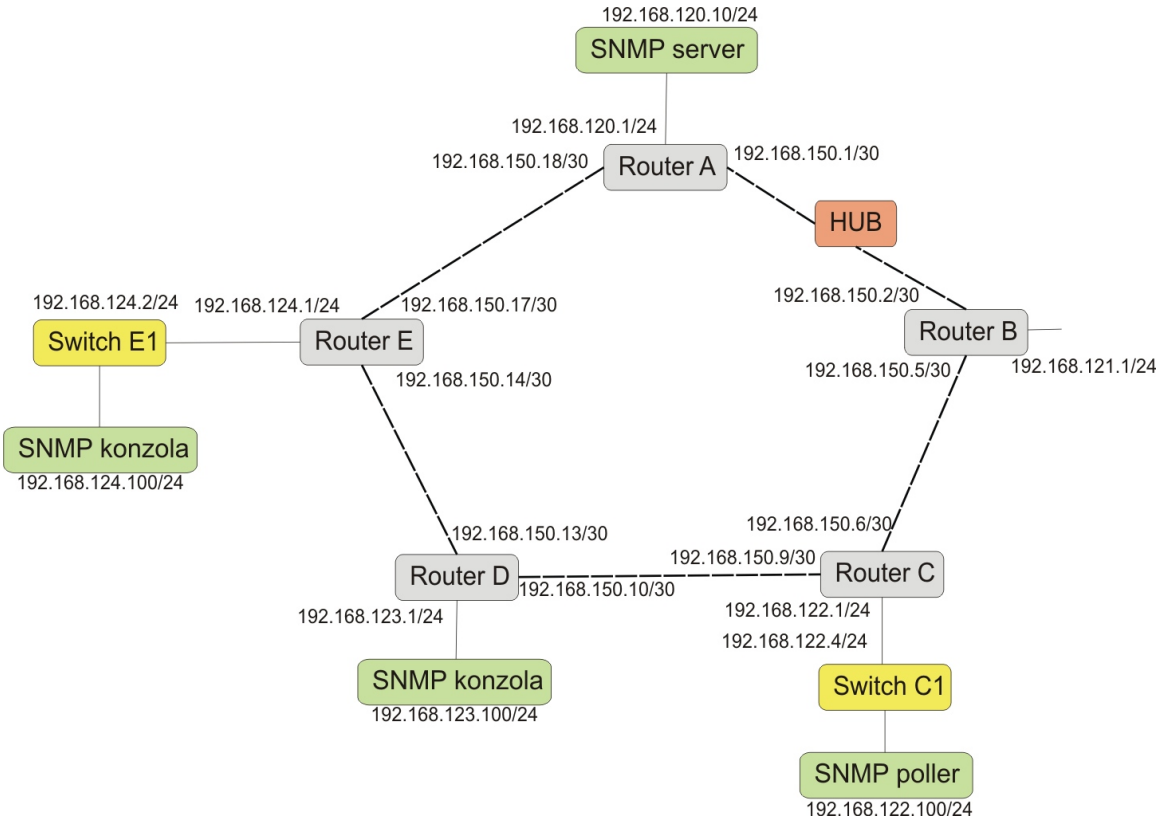
Celý systém SNMPc využívá možností SNMP a je postaven na distribuované architektuře. Je možné do sítě instalovat hlavní a záložní server (tzv. live a standby). Při výpadku hlavního serveru budou automaticky všechny funkce převedeny na záložní. Jedná se téměř o povinnost, kdy se v dnešní době kladou velké požadavky na dostupnost služeb. Celá síť je optimalizovaná tak, aby centrální server nesbíral a nepolloval všechna zařízení sám. Po lince k serveru by tak chodila komunikace s každým zařízením zvlášť a SNMP provoz by byl obrovský. Většina práce je proto rozdělena na lokální servery. Těmito servery se rozumí proxy agenty, nainstalované na zařízeních jako jsou směrovače, nebo přímo pollery. Poller je software SNMPc, který je možné nainstalovat na jakoukoliv stanici (například lokální web server). Tyto pollery budou sbírat data s lokální sítě a nebudou tak zatěžovat páteřní linky ani ostatní sítě. Data mohou samy zpracovávat a odesílat je již v upravené podobě na hlavní server. Na hlavním serveru se také ukládají z bezpečnostních důvodů všechny logy událostí – přihlášení do systému, chybové hlášky, příchozí trapy a podobně. Všechny tyto události jde předem nakonfigurovat tak, aby vyhovovaly administrátorům a nezatěžovaly systém zbytečnými informacemi, ale jen v daný moment důležitými událostmi.

3.2 EXPERIMENTÁLNÍ SÍŤ

V laboratoři PA-427 byla sestavena experimentální síť za účelem ověřování teoreticky nabytých znalostí o problematice správy na reálných síťových zařízeních. Bylo možně si tak

vyzkoušet konfiguraci aktivních prvků i aplikací a přiblížit se tak co nejvíce podmínkám v reálné síti.

Základem sítě bylo pět směrovačů od firmy Cisco serie 1800, což je zařízení vhodné pro malé a střední sítě. Každé z těchto zařízení má dva širokopásmové porty Fast Ethernet WAN (Wide area network) 10/100Base-T Gbit/s. Dále poskytuje možnosti připojení přes V.92 analogový modem - 1811, nebo ISDN S/T BRI (Basic Rate Interface) – 1812. Směrovače jsou také vybaveny 8 porty v zabudovaný přepínači pro lokální síť. Cisco série 1800 umožňuje také vytvoření množství VPN, což jsou virtuální privátní sítě. Zařízení typu 1811/1812 mají řadu funkcí, mezi nimiž je hlavně podpora protokolu SNMPv3, což bylo pro vzdálený dohled a správu těchto prvků nezbytné. Jak je vidět na Obrázek 8, bylo možné s ohledem na dvou portové směrovače sestavit pouze kruhovou topologii. Ta má však několik výhod, které je možné využít pro naše potřeby.



Obrázek 8: Experimentální síť

Do sítě byly také zabudovány dva přepínače (viz Obrázek 8), které rozšiřují naše možnosti pro simulace a umožňují pozdější připojení více hostitelských stanic. Konkrétně se jedná o:

- Switch E1 – HP ProCurve 2626 J4900A – více portový (24) 10/100 Mbps + dva porty podporující 10/100/1000 MBps. Výkonný switch pracující na L2/L3 vrstvě, podporující SNMP.
- Switch C1 – Allied Telesyn AT-8624T/2M – jedná se o propracované zařízení s plnou podporou QoS, pracující na L2/L3/L4 vrstvách. Podpora SNMPv3, OSFPv2, limitování pásma kanálů atd.

V experimentální síti je také hardwarový server Mercury model MP2516HA-R, který slouží jako aplikační základna pro testovací potřeby a tvoří bránu do vnější školní sítě.

Již v takto relativně malé síti by bylo nepraktické užívat statické směrování, proto byl k tomuto účelu vybrán protokol OSPF (Open Shortest Path First). Vzhledem k účelu sítě, kdy

je potřeba neustále měnit topologii a rušit či opětovně nastavovat spoje, je tento protokol ideální. OSFP vyniká oproti RIP (Routing Information Protocol), nebo EIGRP (Enhanced Interior Gateway Routing) hlavně v rychlé konvergenci a jako metriku při výpočtu nejlepší trasy užívá šířku pásma. OSFP je založen na Dijkstrově algoritmu, který vytváří strom nejkratších cest a potom jej zaznamenává do směrovacích tabulek. Zde může být uloženo více cest k jednomu cíli se stejnou metrikou. Protokol také využívá hierarchickou strukturu a je možné síť rozdělit do několika oblastí, avšak v naší síti byla použita pouze jedna oblast – area 0.

OSFP podporuje na rozdíl od RIPv1 také VLSM (Variable Length Subnet Mask). Jak název napovídá, umožňuje adresovat zařízení v síti pomocí variabilní délky sítové adresy. Tato metoda se užívá hlavně pro šetření adresového prostoru, což je dnes při nedostatku IP adress v IPv4 velký problém. Jak je vidět na topologii sítě (viz Obrázek 8) jsou použity síťové adresy třídy C. Síťové adresy v jednotlivých LAN mají prefix /24, což znamená, že tři oktety udávají adresu sítě a posledních 8 bitů adresu stanice, což je 256 adres. Do jedné takovéto sítě je možné připojit až 253 hostitelských stanic. První IP adresa je určena pro název sítě, poslední IP adresa je pro všesměrové vysílání. Konkrétně v experimentální síti je adresa 192.168.X.1/24 použita jako výchozí brána pro sítě LAN. Pro adresaci spojení jednotlivých rozhraní směrovačů bylo využito právě VLSM. Za normálních podmínek s prefixem /24 by bylo pro takový spoj nevyužito 252 IP adres. Pro dvoubodové spojení se ideálně hodí podsíťování s prefixem /30, který umožňuje mít v jedné síti 4 IP adresy, z toho dvě rezervované pro název sítě a všesměrové vysílání a 2 pro jednotlivých rozhraní směrovače.

3.3 INSTALACE PRVKŮ PROGRAMU SNMPC

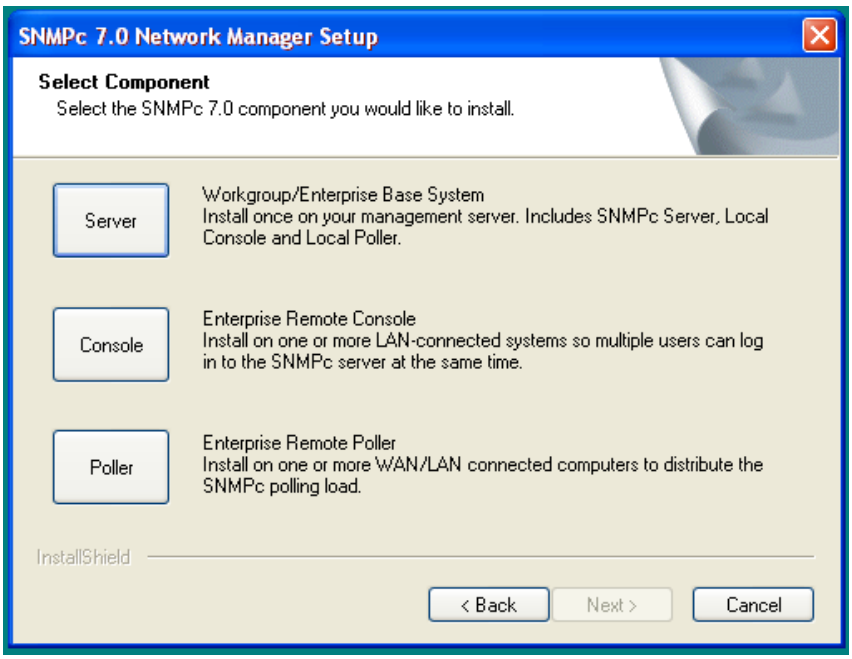
Software SNMPc nabízí tři druhy instalace: server, vzdálenou konzoli, nebo poller. Distribuovaná architektura programu obsahuje všechny tyto prvky v hojném počtu, dle závislosti na velikosti sítě. Logické tedy je, aby u rozsáhlé sítě jeden server mohlo spravovat více administrátorů a také jeden administrátor mohl spravovat více serverů. Na serveru může být v jednom okamžiku připojeno více uživatelů. Jeden přímo na serveru a další na vzdálených konzolách. Pokud pracují současně, změny se projevují ihned.

Licence SNMPc jsou dimenzovány tak, že klasická enterprise edice obsahuje jeden server, deset vzdálených konzolí a deset pollerů. Jestliže zákazník potřebuje více konzolí, nebo pollerů, musí zakoupit rozšiřující balík Remote Access Extension, který umožní využít neomezeně těchto prvků. S Enterprise edition umožňuje správu 25000 zařízení, což je pouze pro deset pollerů příliš. Kdyby každý poller měl dotazovat 2500 zařízení, byla by spotřebována velká šířka pásma pouze pro management sítě a provoz by nebyl efektivní.

3.3.1 Server

Do experimentální sítě byl nainstalován nejprve server SNMPc (NMS). Jako nejvhodnější lokace byl zvolen hardwarový server Mercury s OS Microsoft server 2003 s IP adresou 192.168.120.10 (viz Obrázek 8: Experimentální síť). Při instalaci je nutné zvolit druh instalace jako server (Obrázek 9) a následně vyplnit hodnoty pro počáteční konfiguraci: IP adressu, masku sítě a community string. Community string byl zvolen „lab427“. Je možné buď začít po instalaci rovnou vyhledávat zařízení, nebo zaškrtnou položku *Start with discover off* a později si nastavit podrobnější parametry. Začít s vypnutým vyhledáváním se doporučuje hlavně pro rozsáhlé sítě. Systém začne pomoci ICMP ping detekovat všechny zařízení v síti a je vhodné si nejdříve nakonfigurovat rozsahy IP adres a sektory pro pollery.

Při instalaci bylo nutné myslet také na firewall, který implicitně blokoval porty, na kterých SNMP komunikuje. Pro příjem zpráv trap se jedná o port 162 pro UDP protokol.

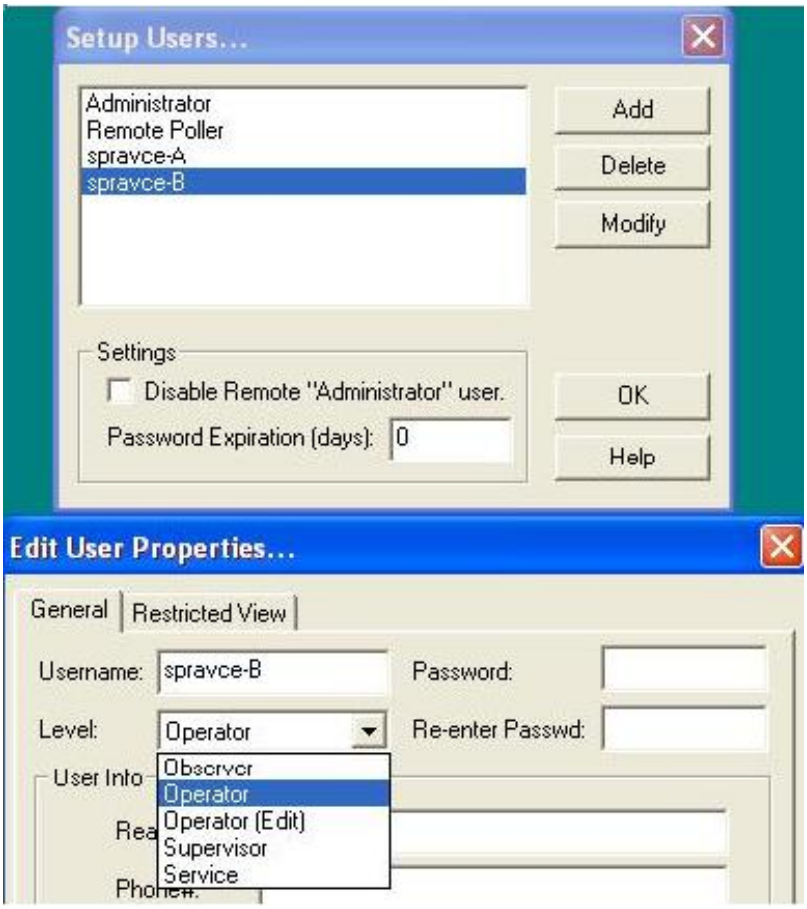


Obrázek 9: Instalace SNMPc

3.3.2 Konzole

Konzoli je možné nainstalovat pouze ve verzi Enterprise nebo v Evaluační verzi. V praxi je vzdálená konzole (Remote console) hojně užívaný nástroj. Pro administrátory by bylo nepraktické, kdyby se směli připojit k serveru pouze z hlavní konzole. Připojení pomocí vzdálené plochy je sice funkční, ale hlavní administrátor nemusí chtít přidělit takováto práva všem dalším správcům. Lze proto využít vzdálené konzole, které se připojí k serveru a chovají se, jako bychom byli připojeni lokálně. Spojení konzole se serverem je provozováno již pomocí protokolu TCP. Při komunikaci serveru s konzolí probíhá velký provoz, protože data jsou aktualizována v reálném čase. To znamená, že změny provedené na konzoli se během velmi krátké doby projeví na serveru. Doporučuje se tedy využívat rychlých spojů přes LAN, nebo linku T1 (E1).

Konzoli je tedy možné nainstalovat na jakoukoliv stanici v síti podporující TCP/IP sadu protokolů. Před samotnou instalací je vhodné na serveru vytvořit účty pro správu sítě a přidělit jim práva. Můžeme tak učinit v záložce *Config/User Profiles*. Pro zajištění co největší bezpečnosti, je vhodné ihned přidělit heslo administrátorskému účtu. Také je zde možné zakázat vzdálený přístup administrátora. Pro jednotlivé segmenty sítě lze určit lokální administrátory a přidělit jim pouze IP adresu sítě, kterou mají spravovat (viz Obrázek 10) a také práva jejich přístupu. Je možné zvolit například účet pouze k prohlížení nashromážděných dat – reportů.

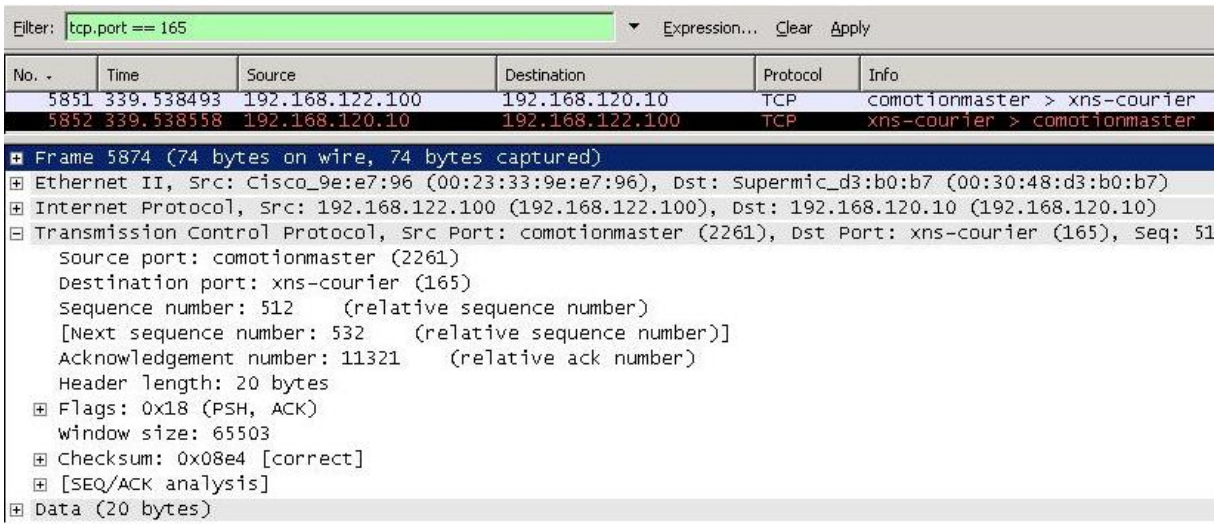


Obrázek 10: Setup User

3.3.3 Poller

Toto zařízení splňuje účel proxy agenta architektury SNMP (viz kapitola 2.1). Konkrétně v případě softwaru SNMPc se jedná o jakousi jednodušší verze SNMPc serveru (NMS), která se umísťuje do vzdálených sítí, které z nějakého důvodu není vhodné monitorovat přímo serverem. Nejčastější důvod je zrychlení monitorování prvků. Pro lepší a rychlejší získávání dat ze sítě. Je logické, že čím více těchto zařízení umístíme do sítě, tím menší nároky budou kladeny na každý jednotlivý prvek, především na spoj mezi SNMPc serverem a monitorovanou sítí. Datový provoz mezi SNMPc manažerem a pollerem nedosahuje oproti konzolám tak velkého objemu, jako kdyby manažer komunikoval přímo s dohledovanými zařízeními, a proto je možné manažer umístit vzdáleně od sledovaných síťových systémů i tam, kde propustnost spoje nižší. Ve vzdálené síti se poller chová jako lokální manažer a monitoruje všechna dostupná zařízení. Na hlavním SNMP serveru je proto dobré přidělit v záložce *Config/Discovery/Polling Agents/Filters* pollerům rozsahy IP adres, kterými se omezí rozsah pollování. Častým problémem je špatné rozdělení úseků sítě mezi pollery, hlavně v případě, kdy je více zařízení v jedné logické části sítě. Různé pollery pak monitorují stejná zařízení a vzniká nadbytečný provoz. Další důvod pro zavádění pollerů je možnost monitorování sítě umístěné za firewallem, nebo proxy serverem. Poller totiž nekomunikuje s NMS pomocí SNMP zpráv, ale přes protokol XNS-Courier (Xerox Network System) běžící na portu 165 protokolu TCP (viz Obrázek 11). V případě, že přepínač ve vzdálené síti obsahující SNMP agenta zjistí výpadek zařízení, odešle místnímu polleru trap – linkDown. Poller tuto informaci zpracuje, vytvoří RPC (Remote Procedure Call) zprávu a zašle ji pomocí XNS protokolu na SNMPc manažera. Manažer naslouchá na portu 165 pro komunikaci

s pollery a tuto zprávu opět přemění na SNMP trap. Takto mohou být všechny RPC zprávy považovány za trap nebo jiné zprávy. Na SNMPc manažerovi pak stačí nastavit pravidla pro klasické SNMP zprávy.



Obrázek 11: Komunikace Polleru s NMS na portu 165

Instalace probíhá obdobně jako u konzole. Na lokálním prvku (stanice, místní server) spustíme instalaci a z nabídky (viz Obrázek 9) vybereme Poller. Vyplníme adresu polleru a lokaci NMS na který bude zasílat data a trapy. Zařízení pracuje automaticky na pozadí systému jako služba. V experimentální síti byl poller nainstalován do sítě 192.168.122.0, jak je vidět ze zachycené komunikace programem Wireshark (viz Obrázek 11). Pro ověření komunikace s NMS, se můžeme podívat v *Config/Discovery/Polling Agents* na hlavní konzolu. Jakmile se poller úspěšně spojí s manažerem, automaticky se nastaví události a zasílání SNMP zpráv ohledně tohoto polleru. Při odpojení zařízení se ihned objeví na konzole alert o ztrátě spojení. Je také nutné na polleru nastavit heslo pro ověření totožnosti a komunikace s manažerem.

3.4 SNMP KONFIGURACE SMĚROVAČE

3.4.1 Základní konfigurace

Zařízení Cisco 1811/1812, která jsou využita v páteřní síti naší experimentální sítě, plně podporují protokol SNMP, jak již bylo uvedeno v kapitole 3.2. Pro správnou funkci je však potřeba provést řadu nastavení, jelikož v továrním nastavení je tento protokol implicitně vypnutý. Konfiguraci je možno provést dvěma způsoby: pomocí webového rozhraní, nebo příkazové řádky - takzvané CLI (Command Line Interface). Užitím webového rozhraní je možné provést opravdu jen ta nejzákladnější nastavení, prakticky pouze protokol SNMP aktivovat v jeho první verzi a určit cílovou adresu pro odesílání zpráv trap. Tyto zprávy se však nebudou odesílat, dokud neprovedeme jejich detailnější konfiguraci v CLI. Je možné tak připojit směrovač do sítě, manažer bude o tomto zařízení vědět a bude ho moci dotazovat dotazy SNMPv1.

Směrovače podporují různé druhy autentizace platné dle užívané verze SNMP:

Verze SNMP	Úroveň	Autentizace	Šifrování
1	noAuthNoPriv	Community String	Ne
2c	noAuthNoPriv	Community String	Ne
3	noAuthNoPriv	Username	Ne
3	authNoProv	MD5 - SHA	Ne
3	authPriv	MD5 - SHA	DES

Obrázek 12: Zabezpečení verzí SNMP

Jak lze vidět, první dvě verze jsou zcela bez zabezpečení, viz 2.3.1. U třetí verze je možné si vybrat z několika možností:

- bez zabezpečení
- přenos SNMP zpráv se zašifrovaným heslem pomocí hashovacích funkcí MD5 (Message-Digest algorithm 5), nebo SHA (Secure Hash Algorithm). Tento způsob nechává možnost neoprávněné analýzy zprávy, avšak poskytuje dohled nad skutečným původcem a zamezuje neoprávněné osobě vydávat se za zdroj SNMP zprávy
- šifrování hesla a zároveň šifrovat zprávu SNMP pomocí symetrické blokové šifry DES(Data Encryption Standard). Což zamezuje neoprávněné osobě přečíst obsah, nebo se vydávat za zdroj SNMP zprávy

Konfiguraci směrovače v CLI je možné provést pomocí programů Hyperterminál nebo Putty. Konstantní příkazy jsou značeny tučně, nepovinné příkazy v hranatých závorkách a proměnné jsou psány kurzívou.

V globálním konfiguračním režimu je možné vyvolat přehled SNMP:
*Router(config)#***show snmp**

Případně:
*Router(config)#***show running-config**

Jako první je vždy nutné aktivovat SNMP agenta. Na směrovači v globálním konfiguračním režimu pomocí příkazu nastavit community string a prává pro čtení a zápis:
*Router(config)#***snmp-server community** *community string* [**RW|RO**]

Dále je nutné určit hosta kterému se budou posílat zprávy trap a notification:
*Router(config)#***snmp-server host** *hostname/IP adress* **version** [**1|2c|3**] *community string*

Následují nepovinné parametry jako lokace ve které se prvek nachází:
*Router(config)#***snmp-server location** *text*

Kontakt:
*Router(config)#***snmp-server contact** *text*

ID zařízení:
*Router(config)#***snmp-server chassis-id** *number*

Velmi důležité je také povolení trap zpráv příkazem:

```
Router(config)#snmp-server enable trap
```

Zde existuje velké množství zpráv které je možné zasílat, například od směrovacích protokolů a podobně. Je vhodné tedy vybrat konkrétní zprávy a zbytečně tak nezatěžovat přenosové pásmo.

Povolení restartu směrovače pomocí snmp:
```
RouterB(config)#snmp-server system shut-dwon
```

Zakázání snmp agenta:
```
RouterB(config)#no snmp-server
```

Je také možné nakonfigurovat zasílání logů přímo ze směrovače a to příkazy:
```
RouterB(config)#logging 192.168.120.10
RouterB(config)#logging trap warnings
```

Kompletní sekvence příkazů například pro směrovač B vypadá takto:
```
RouterB(config)#snmp-server community lab427 RW
RouterB(config)#snmp-server host 192.168.120.10 version 2c
lab427
RouterB(config)#snmp-server location 192.168.121.1
RouterB(config)#snmp-server contact Patala
RouterB(config)#snmp-server enable traps snmp [authentication]
[linkdown] [linkup] [coldstart] [warmstart]
RouterB(config)#snmp-server source FastEthernet0
RouterB(config)#snmp-server source FastEthernet1
```

Tímto bylo zaručeno, že budou na hlavní SNMP server umístěný na IP adrese 192.168.120.10 odesílány vypsané trap zprávy a budou zasílány pouze na porty Fa0 (FastEthernet) a Fa1. Standardně jsou zasílány na všechny porty. Tato funkce se zdá být zbytečná, ale může se stát, že budeme chtít posílat mnoho trap zpráv, nebo bude zařízení generovat velké množství těchto zpráv v případě poruchy. Vlan1 s adresou sítě 192.168.121.0 by tedy byla zbytečně zahlcena těmito zprávami. Navíc tím můžeme zajistit větší bezpečnost proti útočníkovi.

Nakonec si můžeme zkontrolovat nastavení SNMP agenta výpisem:
```
RouterB(config)#show snmp
   Contact: Patala
   Location: 192.168.121.1
   3959 SNMP packets input
       0 Bad SNMP version errors
       18 Unknown community name
       0 Illegal operation for community name supplied
       0 Encoding errors
       7534 Number of requested variables
       0 Number of altered variables
       2261 Get-request PDUs
       1680 Get-next PDUs
       0 Set-request PDUs
       0 Input queue packet drops (Maximum queue size 1000)
   3941 SNMP packets output
```

```
        0 Too big errors (Maximum packet size 1500)
        0 No such name errors
        0 Bad values errors
        0 General errors
        3941 Response PDUs
        0 Trap PDUs

    SNMP logging: enable
        Logging to 192.168.120.10.162, 0/10, 0 sent, 0 dropped.
```

Je možné dále nastavit směrovač jako manažera, nebo rozdělovat uživatele do skupin s jinými právy přístupu. Lze také upravovat délky zasílaných paktů, či specifikovat zasílání zpráv Inform.

3.4.2 Konfigurace SNMPv3 na směrovači

Základní konfigurace SNMP verze 2c je účelná a naprosto vyhovující menším sítím, které jsou oddělené od internetu firewallem, a tam, kde administrátor má přehled o uživatelích a jejich činnostech a kde se nevyskytuje velké bezpečnostní riziko. V rozsáhlých sítích, je však takřka nutností používat zvýšené prostředky pro zajištění bezpečnosti. Kdyby směrovač A v naší experimentální sítí představoval důležitý uzel, spojující sítě dvou oddělení firmy, jakékoliv informace o něm by neměly přijít do nepovolaných rukou. Při používání nezašifrované komunikace může přenášené informace kdokoliv sledovat. Horší případ ale nastane, pokud někdo přečte community řetězec a začne se vydávat za SNMP manažera se stejným řetězcem. Poté získá přístup k prvku a může ho libovolně využívat.

Zavedení třetí verze protokolu do Cisco směrovače a propojení se SNMPc manažerem však není jednoduché. Je nutné vytvořit uživatele a skupiny na směrovači a přidělit jim různá práva pro přístup k SNMP agentovi na tomto zařízení. Na směrovači s názvem RouterA (viz Obrázek 18) proto zavedeme SNMPv3. Budeme předpokládat, že na směrovači ještě nebyl konfigurován SNMP agent.

Pro začátek je nutné aktivovat agenta přiřazením community string a právy přístupu i přesto, že SNMPv3 ke komunikaci vůbec community string nevyužívá:

```
RouterA(config)#snmp-server community lab427 RW
```

Jako první krok provedeme vytvoření uživatele např.: Patala patřící do skupiny: Skupina1. Na výběr jsou tři možnosti šifrování 3. verze (viz Obrázek 12). Parametr *auth* vybírá mezi *sha* a *MD5*. Možnost *priv* zvolí nejbezpečnější formát šifrování hesla i obsahu SNMP zprávy:

```
RouterA(config)#snmp-server user Patala Skupina1 v3 auth sha
psswdlab427 priv psswdlab427
```

Následuje samotné vytvoření skupiny a definování verze a úrovně přístupu pro celou skupinu:

```
RouterA(config)#snmp-server group Skupina1 v3 priv
```

Nyní se přiřadí uživatel k hostovi, což je SNMP manažer přistupující k agentovi na tomto směrovači a úroveň zabezpečení *priv*, kterou bude manažer využívat. Program SNMPc se bude pak moci automaticky spojit s agentem na tomto zařízení:

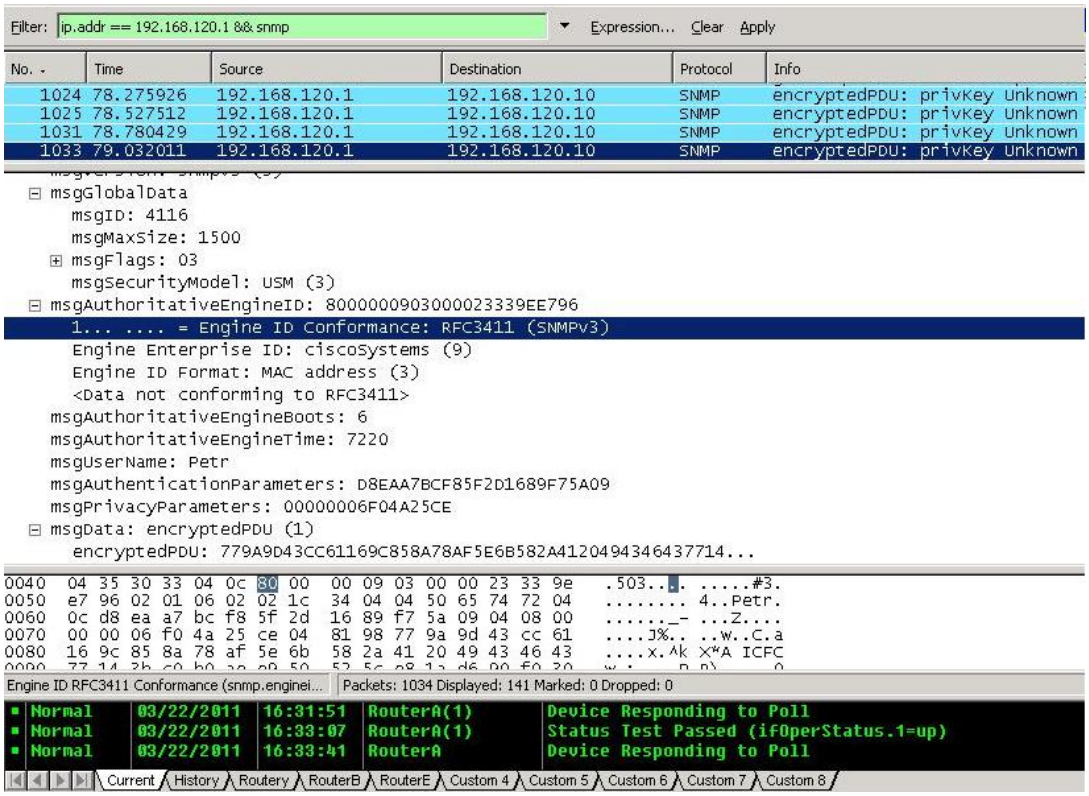
```
RouterA(config)#snmp-server host 192.168.120.10 version 3 priv
Patala
```

Nastavení lokace, kontaktní osoby a dalších parametrů je shodné jako u konfigurace SNMPv2c. Pro administraci uživatelů můžeme využívat příkazy:

```
RouterA(config)#show snmp user
RouterA(config)#show snmp group
```

Dále je možné na směrovači editovat vzdálené uživatele pro přístup k agentovi. K tomu je potřeba editovat engineID - vzdálenou a místní instanci SNMP agenta. Uživatelům je možné zase ve skupině přiřazovat zvláštní práva pro čtení a zápis a zasílání oznamovacích zpráv a podobně. Vzhledem k tomu, že pro tyto účely využíváme program SNMPc, není potřeba tyto věci složitě nastavovat na směrovači. Je nutné pouze pro prvotní zabezpečený kontakt manažera s agentem a po ověření lze již editovat hesla a ostatní funkce z manažera.

Při zachycené komunikace programem Wireskark (viz Obrázek 13) je možné vidět, že již nerozeznáme ani soukromý klíč, ani data přenášená ve zprávě. Přečíst lze pouze část hlavičky, která však bez zbylých dat není k ničemu.



Obrázek 13: Wireshark SNMPv3

3.5 SNMP KONFIGURACE PŘEPÍNAČE

V experimentální sítí byl použit mimo jiné přepínač Allied Telesyn AT-8624T/2M, plně podporující SNMPv3. Přepínač obsahuje podporu QoS a POE (Power Over Ethernet) Tento 24 portový L3 switch byl umístěn do podsítě C s IP adresou 192.168.122.4. Přepínač nabízí možnost celkem kvalitní konfigurace přes webové rozhraní, avšak pro naše účely konfigurace SNMP je potřeba opět sáhnout do příkazové řádky.

Samotná konfigurace se provádí odlišně, než u zařízení Cisco. Všechny příkazy se píší do základního režimu pouze s různými parametry. Jako první povolíme SNMP na přepínači, agent je při implicitním nastavení vypnutý a nebude odpovídat na žádné SNMP dotazy:

```
SwitchC1>enable snmp
```

Obdobně jako u směrovače je nutné vytvořit communitu s příslušnými přístupovými právy tímto nastavením aktivujeme možnost komunikovat se zařízením pomocí SNMPv2c, pro třetí verzi je konfigurace odlišná:

```
SwitchC1>create snmp community=lab427 acces=write
trapnost=192.168.120.10 manager=192.168.120.10 open=no
v1traphost=192.168.120.10 v2ctraphost=192.168.120.10
```

Je možné vytvořit další komunity (pomocí klíčového slova Add), nebo tuto komunity zrušit (Destroy). Dále povolíme zasílaní zpráv trap pro danou komunitu:

```
SwitchC1>enable snmp community=lab427 trap
```

V tento moment nebude zatím zasílání trap zpráv funkční. Zprávy typu linkUP a linkDown jsou ve výchozí nastavení zakázány. Jejich zasílání je nutné aktivovat na každém portu např:

```
SwitchC1>enable snmp interface=1 linktrap
```

Nastavíme si také zasílání maximálně dvou zpráv za minutu:

```
SwitchC1>enable interface=1 traplimit=2
```

Pro konfiguraci SNMPv3 je potřeba vytvořit uživatele a skupiny, jako v případě konfigurace směrovače. Je možné také vytvořit zvláštní práva pro čtení, nebo zápis a oznamování (tzv. view) a ty pak zvlášť přiřazovat konkrétním skupinám, ale v našem případě by to neplnilo účel. Úroveň přístupu je volena AuthNoPriv, čili zpráva je autentizována pomočí hashovací funkce, ale je možné ji přečíst:

```
SwitchC1>add snmp group=admin securitylevel=authnopriv
```

Vytvořením uživatele dané skupiny, uživatel sice nemusí používat privátní heslo, jelikož je pouze ve skupině pro AuthNoPriv přístup, ale mohl by být součástí více skupin:

```
SwitchC1>enable snmpuser=Patala group=admin authprotocol=MD5
authpassword=psswdlab427 privprotocol=DES
privpassword=psswdlab427
```

Nyní je potřeba specifikovat cíl a parametry SNMPv3 komunikace:

```
SwitchC1>add snmp targetparams=netmonpc
securitylevel=authnopriv user=Patala

SwitchC1>add snmp targetaddr=192.168.120.10 UDP=162
params=netmonpc
```

Kontrola nastavení přepínače probíhá příkazem s následným výpisem:

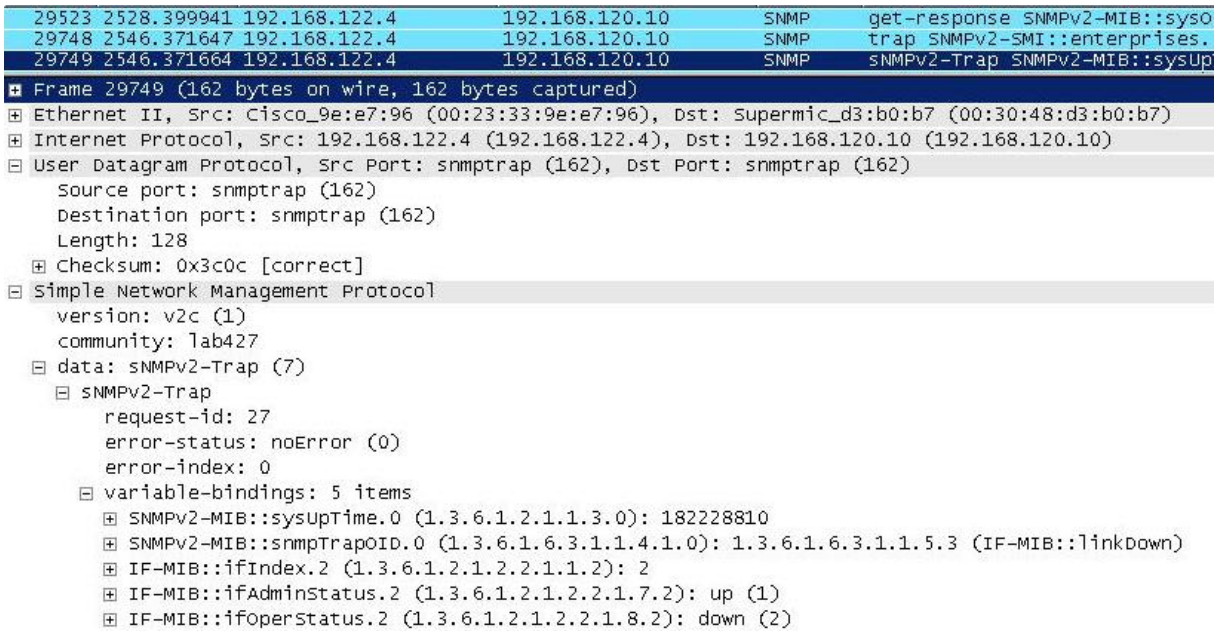
```
SwitchC1>show snmp community=lab427
```

Výstup bude v našem případě vypadat takto:

```
   SNMP community information
     Name ............................ lab427
     Access .......................... read-write
```

```
Status ..........................  Enabled
Traps ...........................  Enabled
Open access .....................  No
Manager .........................  192.168.120.10
Trap host .......................  192.168.120.10
V2c Trap host ...................  192.168.120.10
```

K otestování správné funkce synchronizace toho přepínače a SNMPc manažera použijeme program Wireshark. Jestliže vypojíme jakékoliv zařízení z portu přepínače, měl by zaslat trap na výše uvedenou adresu. To, že ho skutečně zaslal, vidíme na Obrázek 14. V poli variable-bindings vidíme stav spoje ifindex.2, což je port Fa 2 na přepínači. Port je správně nakonfigurován, ale není k němu připojeno zařízení, jelikož došlo k jeho odpojení (snmpTrapOID - linkDown). Na mapě objektů SNMPc konzole je také toto zařízení viditelné s popiskem SNMP, jelikož nyní podporuje tento protokol. Při odpojení zařízení z jeho portu, se objeví také ohlašovací okno *SNMP NT Alarms*...

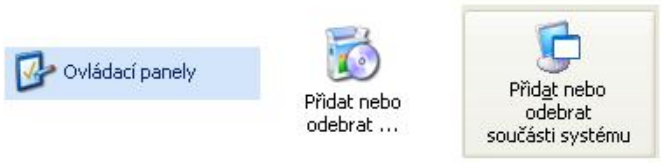


Obrázek 14: Komunikace přepínačem s NMS

3.6 SNMP KONFIGURACE KONCOVÉ STANICE

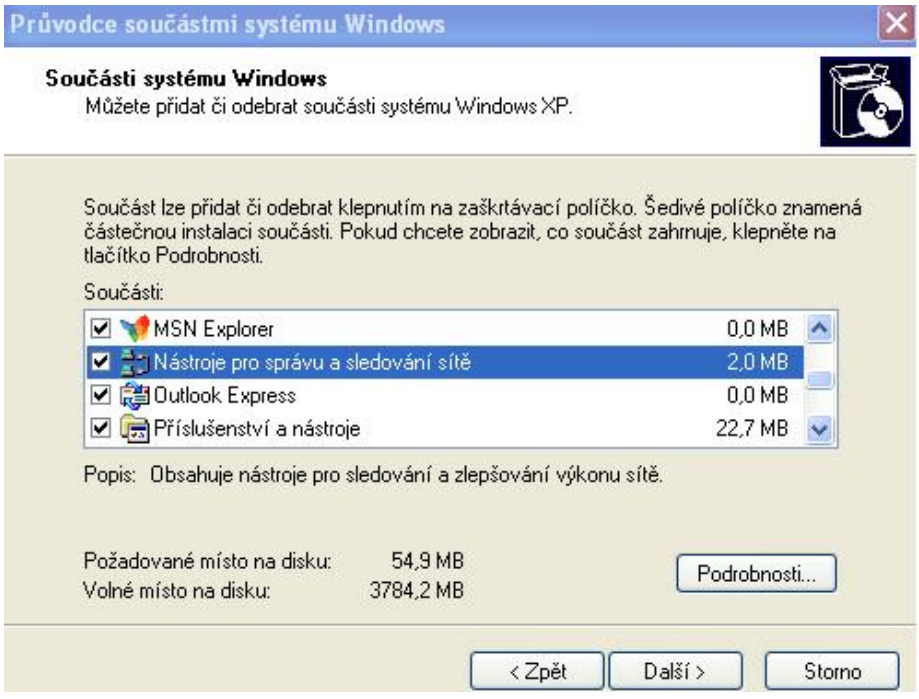
Pro administrátora je také důležité mít přehled o jednotlivých koncových stanicích. Stanice s operačním systémem Windows je možné monitorovat pomocí proprietárního řešení Microsoftu. Existuje také možnost nainstalovat na stanici agenta pomocí freeware programů, jako je například **Snmpv3agent**. Tyto bezplatné verze však obsahují pouze malé možnosti dohledu, většinou pouze aplikační a síťové vrstvy.

Operační systémy Windows XP a starší, neobsahují implicitně při instalaci balík SNMP, ale je možné jej doinstalovat. Nabídka *Start* / *Ovládací panely*, dále pak *Přidat nebo odebrat programy* a v levé části *Přidat nebo odebrat součásti systému*.(viz Obrázek 15).



Obrázek 15: Instalace SNMP na Win XP – 1

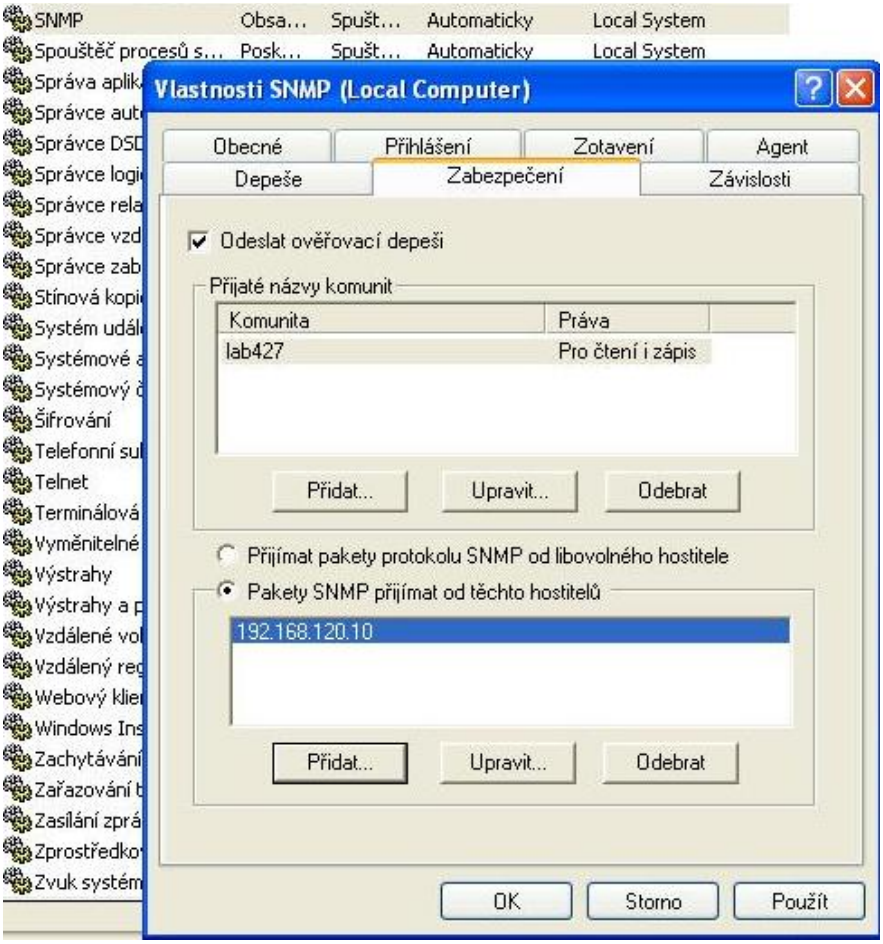
Dále pak v okně *Průvodce součástmi systému Windows* zvolíme *Nástroje pro správu a sledování sítě* (viz Obrázek 16) a vybereme protokol SNMP. Po instalaci je agent zatím deaktivován.



Obrázek 16: Instalace SNMP na Win XP – 2

Pro jeho aktivaci spustíme nabídku *Start/Spustit /Services.msc*. V tomto okně vyhledáme SNMP agenta a následně můžeme ve vlastnostech provádět jeho nastavení. V záložce depeše lze nastavit cíl odeslání zpráv Trap. V záložce zabezpečení pak community name a rozsah adres, od kterých bude agent přijímat dotazy (viz Obrázek 17). V menu Agent, pak můžeme zadat kontaktní informace o agentovi a vybrat vrstvy modelu TCP/IP, které se budou

monitorovat (podrobněji viz kapitola 4.7). Co je důležité vědět, že ani Windows Vista a nižší, ani Windows server 2003/2008 nepodporuje SNMPv3, pouze v2c. Což je velká bezpečnostní díra. Pokud by chtěl administrátor síť udržet maximálně bezpečnou, musel by využít nabídky na instalaci agenta na Windows od jiných firem, například **LoriotPro** s podporou SNMPv3.



Obrázek 17: Instalace SNMP na Win XP – 3

Jestliže na stanici běží SNMP agent od Windows a chceme na něj nainstalovat třeba poller SNMPc, či jinou SNMP součást, je vhodné agenta vypnout a nastavit spouštění služby manuálně. V opačném případě se mohou vyskytnout komplikace a služby by nemusely fungovat správně.

4 MONITOROVÁNÍ A SPRÁVA EXPERIMENTÁLNÍ SÍTĚ

4.1 ZÁKLADNÍ NASTAVENÍ

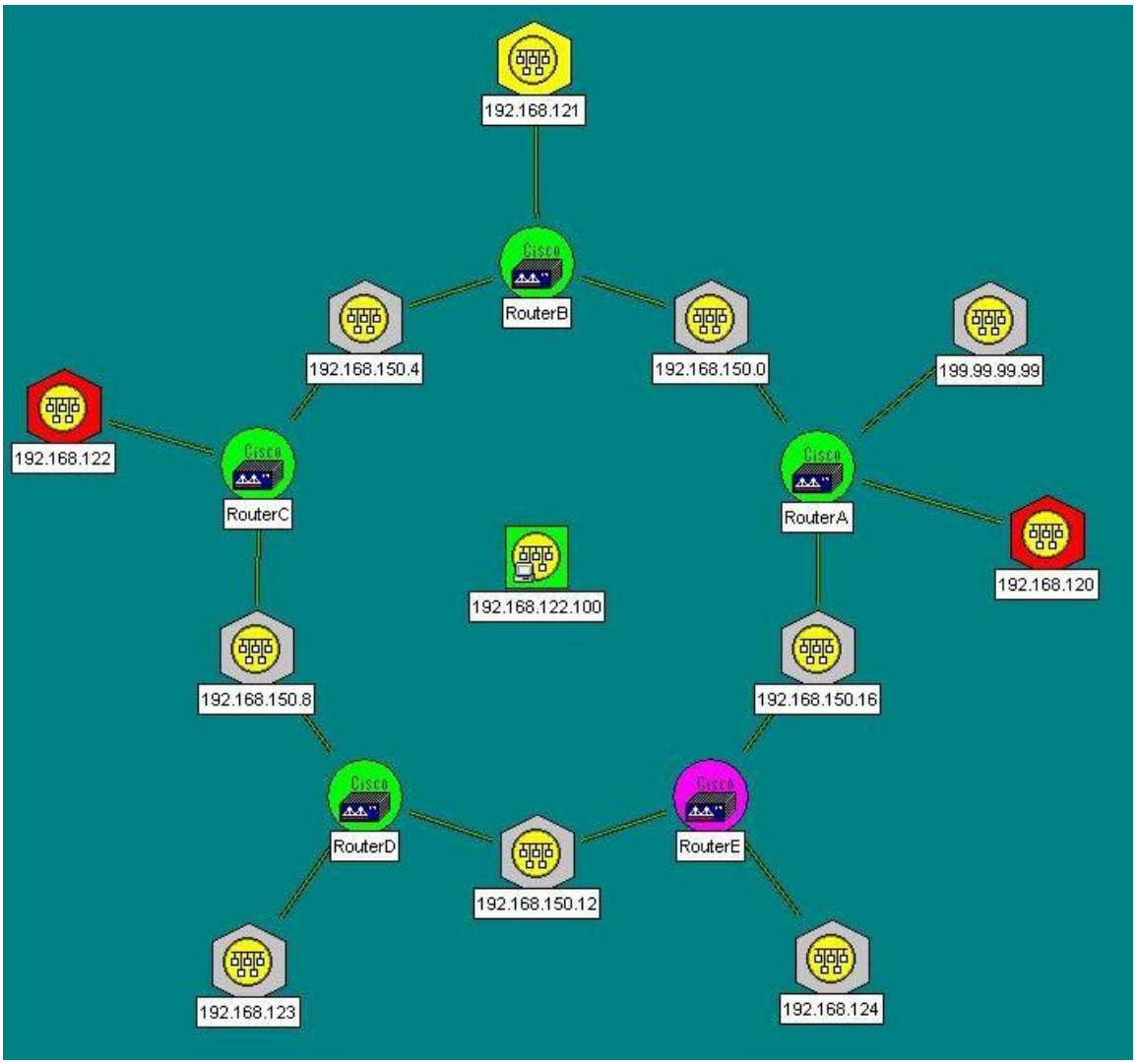
Všechny prvky programu SNMPc byly nainstalovány do experimentální sítě. Všechny směrovače byly nakonfigurovány k přijímání SNMP dotazů od manažera a zasílání zpráv trap zpět k manažerovi. Abychom ověřili správnost těchto konfigurací začneme monitorovat síť. Na SNMPc serveru v záložce *Config/Discovery/Polling Agents*. Před vyhledáváním je vhodné rozdělit kompletní topologii na menší části a přiřadit ji k jednotlivým pollerům. Experimentální síť obsahuje pouze jeden poller v síti 192.168.122.0. Proto v záložce filtr upravíme vyhledávací rozsahy tak, aby se vzájemně nepřekrývaly s manažerem. Síť 192.168.122.0 bude tedy simulovat vzdálenou síť umístěnou za firewallem, nebo proxy serverem. Zprávy trap na portu 162 ze směrovače C by tedy mohly být na některém z mezilehlých firewallů zakázány (viz kapitola 3.3.3). Proto je výhodné přesměrovat tyto zprávy na poller umístěny v této síti na stanici 192.168.122.100. Poller se na mapě objektů zobrazuje jako samostatná ikona (viz Obrázek 18). Po rozkliknutí je možné sledovat stav sítě tak, jak ji vidí právě toto zařízení. Sám poller se chápe jako místní SNMPc server a topologii uspořádává podle toho.

Provedeme další nastavení v záložce *General/ Enable discovery*. Tímto povolíme manažerovi pravidelně dotazovat zařízení a vyhledávat nová. Povolíme i *Ping Scan Subnets*. Tímto bude manažer pomocí ICMP Ping pravidelně dotazovat zařízení ve všech sítích, a tak zjišťovat jejich konektivitu. Zapneme si i ověřování služeb jako je www, Telnet a FTP v záložce *Proto* a umožníme systému nalézt i zařízení nepodporující SNMP *Find Non-SNMP (Ping) Nodes*.

Základ, pro úspěšnou komunikaci manažera se SNMP prvky je také nastavení community řetězce (2.3.2) v záložce *Comm*. Základní přednastavení public a netman smažeme a vložíme nové pro čtení i zápis. V laboratoři je na všech zařízeních užit řetězec lab427. Pro přehlednost a přiřazování nových zařízení můžeme nastavit v menu Layout:

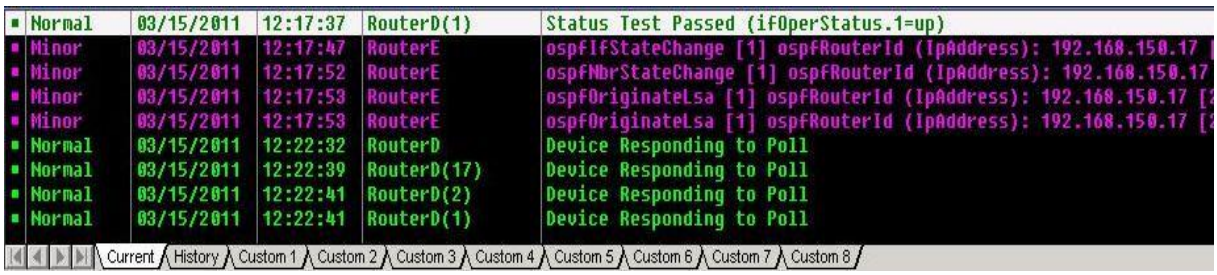
- Top Level/Complete – celá mapa se nám bude automaticky měnit, podle nových nalezených prvků.
- Top Level/Incremental - nové prvky se nám budou automaticky přiřazovat, ale staré zachovají svoji polohu.
- Discovery Objects – nové prvky se budou objevovat v nové síti Discovery Objects a administrátor je pak bude přiřazovat manuálně.

Vyhledávání zahájíme tlačítkem restart a během několika málo minut se postupně zobrazí kompletní mapa sítě. Za zmínku stojí také fakt, že po připojení a následném odpojení stanic budou jejich ikony stále na mapě označené jako Down, dokud neprovedeme jejich manuální odstranění, nebo restart celé mapy. Experimentální síť má kruhovou topologii – všechny spoje jsou funkční. Obrázek 18 zobrazuje aktuální stav síťových prvků jak v grafické podobě. Barvu jednotlivých prvků ovlivňuje jejich stav, lze ji manuálně nastavovat a měnit parametry v záložce *Properties* při výběru prvku. Stav prvků se mění podle příchozích událostí na manažerovi. V implicitním nastavení stanice, která se nehlásí, dostane na dobu 2 x 30 sekund žluté označení, poté přechází do stavu Down – červená. Zařízení, které zaslalo trap zprávu, je označeno fialově atd. Všechny tyto parametry lze měnit a je výhodné natavit vlastní reakce na události (eventy).



Obrázek 18: SNMPc Experimentální síť

Pro lepší přehled o tom, co se ve skutečnosti děje na síti byl Router E využit jako testovací směrovač pro zasílání více druhů trapů. Povolení trap zpráv v CLI směrovače nastavíme příkazem: *RouterE(config)#snmp-server enable traps*. Tímto ale způsobíme zahlcení manažera obrovským množstvím zpráv, které budou zahlcovat síť. Proto je účelnější vybrat jenom ty, které budeme opravdu potřebovat. Pomocí nápovědy *?* lze zjistit všechny možnosti příkazu *enable traps*. Pro názornost: *RouterE(config)#snmp-server enable traps ospf* povolíme pouze zasílání zpráv týkající se směrovacího tohoto protokolu. Obrázek 19 zobrazuje logovací okno programu SNMPc, které zaznamenává právě příchozí OSPF trapy z IP adresy 192.168.150.17, což je rozhraní směrovače E připojené ke směrovači A.



Obrázek 19: OSFP zprávy trap

Program SNMPc umožňuje připojit se přes telnet ke všem zařízením podporující tento protokol. Stačí označit toto zařízení a vyvolat pravým tlačítkem myši nabídku *Tools/Connect Telnet*.

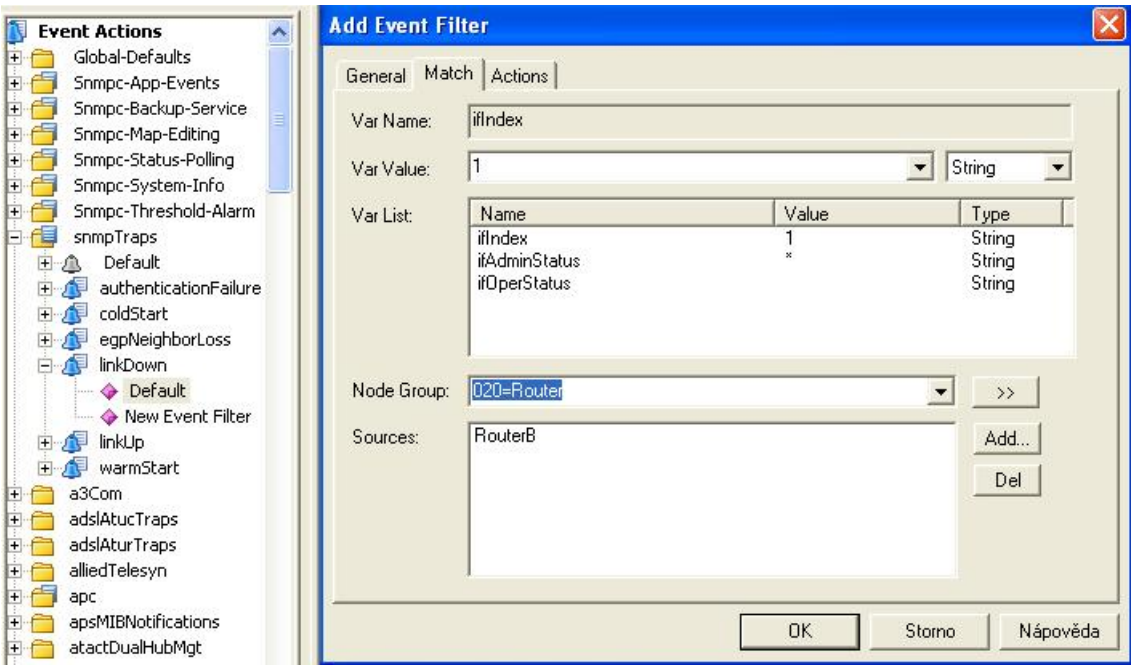
4.2 VÝPADEK SPOJE A JEHO OHLÁŠENÍ

Jedním z nejzávažnějších problémů vlastníka sítě je výpadek spoje mezi dvěma zařízeními, ať už je důvod jakýkoliv (fyzické poškození kabelu, či nesprávná konfigurace spoje). Tento problém je nutné ihned lokalizovat a následně řešit, buď fyzicky nahrazením vadného kabelu, nebo přesměrováním provozu dočasně přes záložní hardware. Program SNMPc umožňuje po správném nastavení velmi rychlou lokalizaci problému v rámci několika sekund. Aby tyto procedury byly, co nejvíce viditelné byl upraven interval směrovačů z 30 na 300 sekund. Toto nastavení je možné provést výběrem všech směrovačů a pravým klikem v nabídce *Properties* . Nejdříve v záložce *Generals* přiřadíme směrovače do skupiny *020=Routers* a poté v záložce *Access* lze změnit interval pollingu. Tímto bylo zajištěno, že systém bude dotazovat zařízení po pěti minutách a skutečné zjištění závady pomocí zpráv trap bude lépe viditelné. Pokud by zprávy trap nebyly aktivovány, SNMP manažer by v nejhorším případě rozeznal výpadek spoje až po uplynutí tohoto intervalu. Nejdříve by označil zařízení za neaktivní a jako zařízení s poruchou ho označil až po uplynutí dalších nastavitelných intervalů. Celkově by takováto doba odezvy sítě na poruchu byla nepřijatelná.

Jestliže máme nyní nastaveny směrovače na odesílání zpráv trap, po rozpojení kabelů mezi směrovači C a D se během několika málo sekund v logu událostí objeví informace o poruše. Do tohoto okna se ale zapisuje velké množství informací a pro administrátora by zachycení této poruchy bylo obtížné. Lze proto pro každou situaci nastavit akci, která se má vykonat – takzvaný event – a odlišit tak nepodstatné události od závažných.

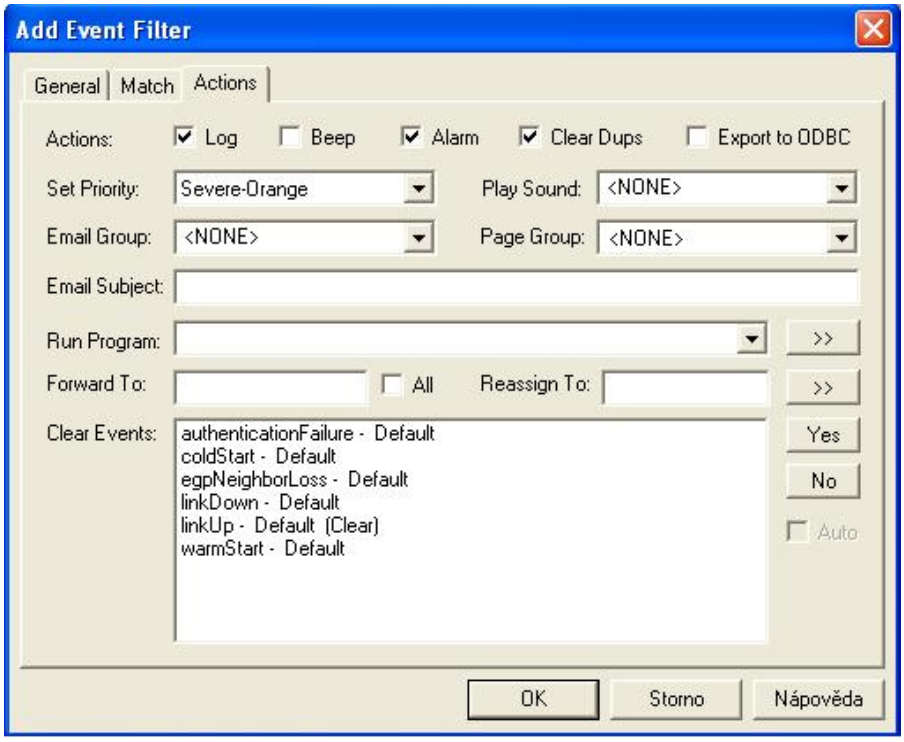
Pro naši experimentální síť je nejdůležitější monitorovat konektivitu mezi směrovači, to je porty FastEthernet (zkráceně Fa) 0 a 1. Naopak nechceme, aby se ohlašovala přihlášení jednotlivých pracovních stanic na ostatních portech. Základní nastavení pro výpadek jakéhokoli spoje je ohlášení a červené zbarvení směrovače. V záložce Event v levé části konzole vybereme *Event Action/snmpTraps/linkDown*. Upravíme implicitní nastavení pro všechny akce po příchodu zprávy trap linkDown - *Properties/Actions*. Zde provedeme změnu priority na *Normal-Green*. Nyní se nám žádné výpadky linek na portech směrovače nezobrazují červeně, ale stále se logují. Zařízení tak hlásí normální provoz při odpojování a připojování zařízení k jeho portům.

Pro ohlašování výpadku důležitých portů je nutné vložit vlastní událost. Označíme předem vytvořený event v záložce *Event Action/snmpTraps/linkDown* a pravým kliknutím vložíme nový *Insert Event Filter*. Zkopíroval se nám základní filtr se všemi atributy, ten teď můžeme upravovat. Nazveme ho například Int 1. Do kolonky *Message* je možné napsat to, co se bude zobrazovat v logu událostí. Pomocí parametrů jako $T (zobrazí community name zprávy trap), $B (MAC adresa) lze umožnit lepší orientaci v logu. V záložce *Match* (viz Obrázek 20) nastavíme parametry, které musí souhlasit se zachycenou zprávou trap, aby se vyvolal tento event. V tomto případě klikneme do pole *Var List* hodnota *ifIndex* a do pole Var Value nastavíme parametr 1. Znamená to, že pokud bude číslo portu 1, provede se tato událost.

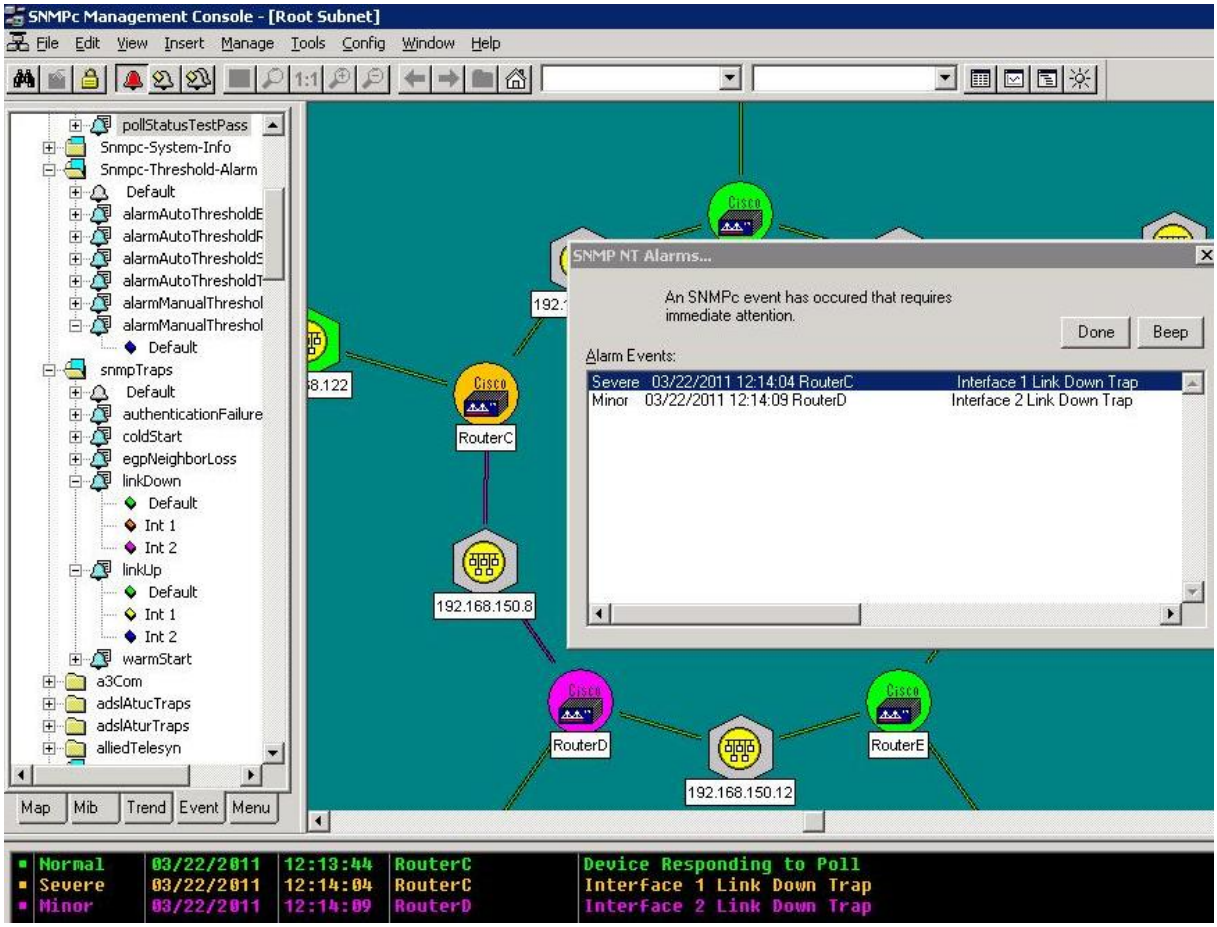


Obrázek 20: Add Event Filter - Match

Také vymezíme, pro která zařízení tato událost bude platit v listboxu *Node group*, kde zvolíme *020=Routers* skupinu všech směrovačů. Nyní v záložce *Actions* nadefinujeme, co se stane. Změna barvy a priority na *Severe-Orange*, zaškrtnutí políčka *Alarm* v horní části, pro oznámení přímo administrátorovi. V poli *Clear Events* označují eventy, které se potvrdí (Acknowledge) v případě této události v tabulce aktuálních probíhajících událostí. Vytvořením podobného eventu pro linkDown na portu 2 a linkUP na portu 1 a 2 dosáhneme přehledného oznamování.



Obrázek 21: Add Event Filter - Actions



Obrázek 22: Trap linkDown

Po nastavení všech událostí (viz Obrázek 22 a Obrázek 23) je možné sledovat průběh oznamování a logovaní výpadku trasy mezi směrovači C a D. Směrovače hlásí alarmy do okna SNMP NT Alarms, které se zobrazí administrátorovi na konzole. Toto okno je nutné manuálně potvrdit, jinak se nezavře. Postupně přicházejí alarmy v čase:

- 12:14:04 RouterC zasílá trap (oranžový alarm), hlásící výpadek rozhraní Interface1 (SNMPc nekoresponduje v označování se skutečným názvem portu na směrovači, čili Fa 0 je v SNMPc Interface 1 atd.).
- 12:14:09 RouterD zasílá trap (růžový alarm), hlásící výpadek rozhraní Interface 2
- 12:20:14 RouterC hlásí nahození Interface 1 (žlutý alarm)
- 12:20:15 RouterD hlásí nahození Interface 2 (modrý alarm)

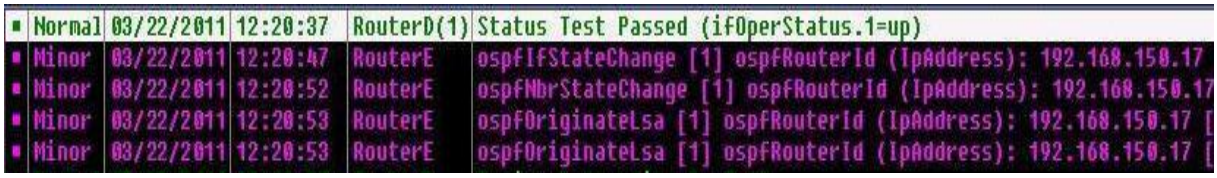
Obrázek 23: Trap linkUp

Dále je nutno ošetřit události, které budou potvrzovat další události. SNMPc nabízí možnost zobrazení kompletní historie událostí, nebo jednotlivých skupin či samostatných zařízení. Ve spodní části konzole je možné tyto události zobrazovat v záložkách *Current*,

History, nebo nastavit vlastní záložky. Nebo také zobrazovat po kliknutí na jednotlivá zařízení. Aktuální stav zařízení ovlivňují právě aktuální události a jejich potvrzování (nastavení *Clear Events* viz výše).

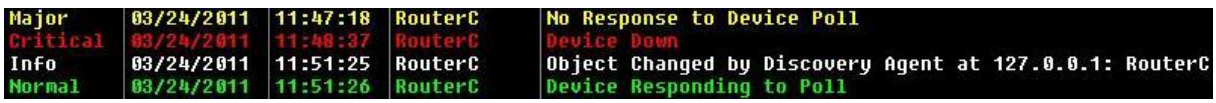
Například směrovač C při zahlášení výpadku Interface (zkráceně Int) 12 přejde do stavu linkDown (po nastavení stále priorita Normal). Při nahození Int 12 přejde do stavu linkUp (priorita Normal). To znamená, že tyto události na sebe na vzájem ukazují hodnotou *clear* (viz Obrázek 21). Událost linkDown Int 1 přehodí prioritu na Severe (oranžová) a potvrdí jak událost linkDown (defaultní), tak událost linkUP Int 1. Opačně bude zpráva linkUP Int 1 potvrzovat jak linkDown, tak linkDown Int 1. Analogicky nastaveno pro Int 2. Tímto je docíleno toho, že výpadky rozhraní 1 a 2 změní prioritu zařízení, výpadky ostatních rozhraní ponechají prioritu takovou, jaká byla. Přitom si výpadky rozhraní 1 a 2 navzájem nebudou měnit (potvrzovat) priority. Jinak by se totiž stalo, že kdyby vypadly Int 1,2 a 10, pak by nahození Int 10 (nebo 2) změnilo prioritu na normal a administrátor by viděl všechno v pořádku i přes to, že Int 1 by bylo stále down.

Provoz je směrován po výpadku trasy přes zbývající část sítě, o čemž nás také informují trapy o změně topologie protokolu OSPF.



Obrázek 24: Změna topologie OSPF na směrovači D

Problém nastává při výpadku spoje mezi směrovači, které jsou oba blíže jedné větvi. Rozpojíme-li například směrovače B a C, potom ze směrovače B dostaneme trap o výpadku rozhraní 1. Od směrovače C však žádné oznámení nedostaneme, i když by teoreticky měla zpráva najít cestu přes směrovače D a E. Problém je v tom, že manažer SNMPc vidí každý směrovač jako zařízení, ke kterému je přiřazena jen jedna IP adresa. Konkrétně směrovač C má přiřazenu IP adresu 192.168.150.6 na portu 2. Toto rozhraní ale přestane fungovat, a tudíž směrovač pošle trap z rozhraní 1 IP 192.168.150.9. SNMPc však toto rozhraní nepřiřadí ke směrovači C, protože ho má uloženo pod původní IP adresou. K nápravě dojde až při další periodě *Auto Restart Time* (výchozí nastavění je 1 hodina) v záložce *Discovery/Polling Agent*. Po uplynutí této doby je směrovači C přiřazena nová IP adresa dostupného rozhraní 1. Konkrétní časové hodnoty o informovanosti manažera závisí na tom, do jakého úseku periody se právě výpadek trefí. V jednom případě byl výpadek zaznamenán příchodem trapu ze směrovače B v 11:48 a SNMPc manažer opravil již v 11:51 IP adresu směrovače C, discovery agent ohlásí změnu do logu událostí (viz Obrázek 25). Při dalším pokusu byl výpadek linky zaznamenán v 13:45 a oprava byla provedena až o 40 minut později v 14:24, což už je poměrně dlouhá doba. Než je provedena oprava, manažer po uplynutí pollovacího intervalu a vypršení časovače o nedostupnosti označí zařízení jako down. Nemá k němu přístup ani přes telnet (ten přistupuje k rozhraní, které je shozené) ani není možné přistupovat přes SNMP protokol a pollovat z něj data. Dočasným řešením je manuálně ve vlastnostech zařízení změnit IP adresu na některou z funkčních portů.



Obrázek 25: Automatická změna ip adresy portu směrovače na SNMPc manažerovi

4.3 MĚŘENÍ A ZÍSKÁVÁNÍ DAT

4.3.1 Dlouhodobé statistiky

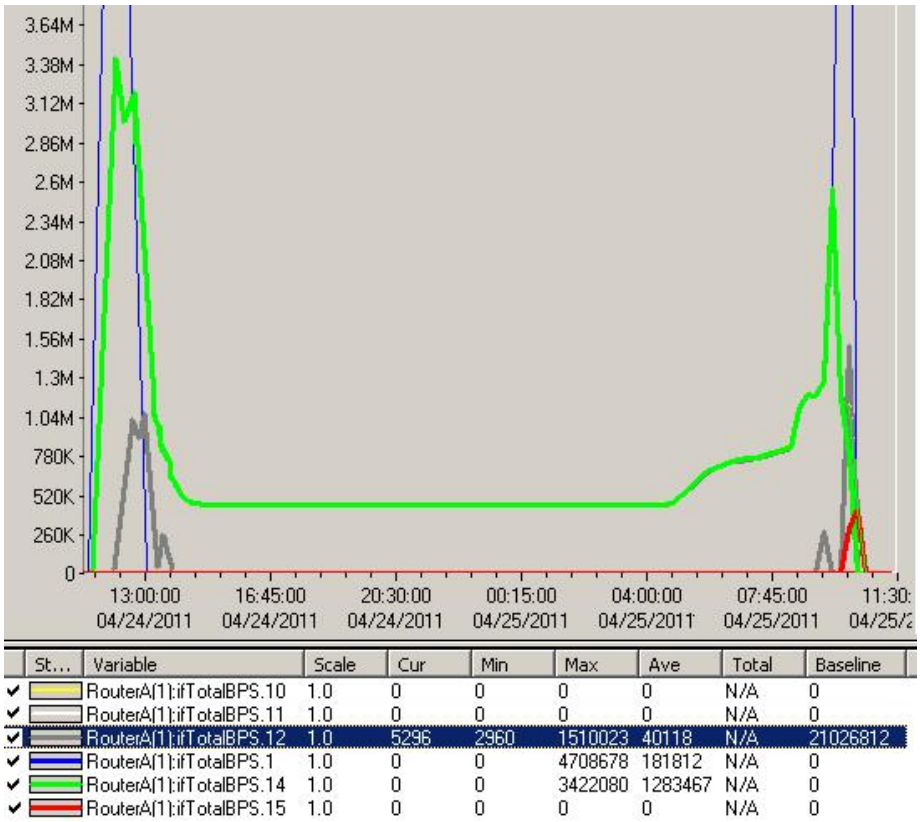
Program SNMPc nabízí spoustu možností jak monitorovat síťová zařízení. Stahováním dat v podobě tabulek MIB lze získat informace o kterémkoliv zařízení podporující SNMP protokol (viz 2.3.3). Tyto informace je možné získávat v pravidelných intervalech a poté z nich vytvořit graf. Jedná-li se například o informace o průběžném vytížení portů nebo stav paměti serveru, může nám graf poskytnout přehlednější možnost sledování než tabulka. Nejzákladnější informace je možné získat po pravém kliknutí na zařízení a v patřičné záložce vybrat akci. U směrovače podporujícího protokol SNMP jsou to informace o portech, jejich momentální přenosové rychlosti, směrovací tabulky, informace přímo o technologiích Frame Relay, ATM atd. Prohlížením těchto informací však mnoho informací o dlouhodobějším chodu sítě nezískáme. Měření aktuálních parametrů sítě je funkce, která dokáže s velkou přesností analyzovat její stav. Základním intervalem pro toto měření je 10 sekund (minimálně 1 s), proto je toto měření velmi podrobné. S tím je však spojena také zátěž sítě, která je tímto měřením zvyšována. Dlouhodoběji je proto lépe tuto funkci nevyužívat ve velké míře.

Ideálním nástrojem SNMPc jsou pro tyto účely dlouhodobé statistiky (Trend Report Long Term Statistic). Tyto statistiky je možné získat v předem nastavitelných intervalech. Manažer automaticky nakonfiguruje agenty, aby co nejefektivněji získávaly data a zasílaly je k vyhodnocení na NMS. Základním intervalem pro získání dat je 10 minut, minimální pak 1 minuta pro detailní měření. Takto získané výsledky je možné ukládat do grafů zobrazujících hodinové, denní, týdenní, měsíční charakteristiky.

Výsledkem měření takovéto statistiky může být graf podobný jako na Obrázek 26. Zde je jednodenní statistika ze dne 24. dubna 2011 posunutá v časové ose o několik hodin tak, aby zobrazovala celý průběh měření přenosů dat na směrovači A. Tyto statistiky je možné zoomovat a posouvat tak, aby administrátorovi mohly poskytnout pohled na kteroukoliv požadovanou dobu.

Samotné vytváření statistik probíhá v záložce *Trend* v levé části konzole. Všechny statistiky jsou umístěny do přehledného stromu *Trend Report*. V tomto stromě lze také vytvářet skupiny reportů, které budou moci sledovat jen uživatelé s různými právy pro přístup a zamezit tak prohlížení všech statistik nepovolanými uživateli.

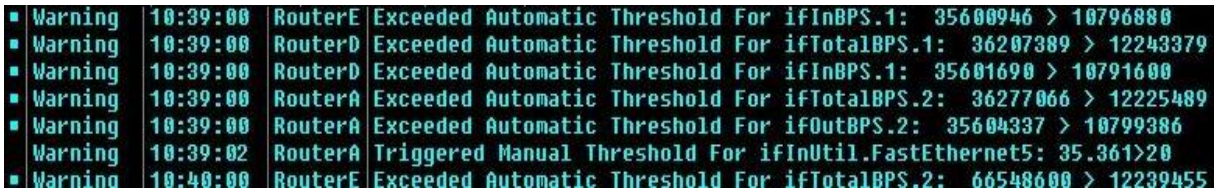
Podrobněji je vytvoření dlouhodobých statistik popsáno v kapitole 4.3.3, kde je rozebrána problematika vytváření prahových hodnot a vytváření filtrů na události, při kterých dochází k překročení právě těchto prahových hodnot. Obě kapitoly tedy úzce souvisí.



Obrázek 26: Dlouhodobá statistika přenosu dat

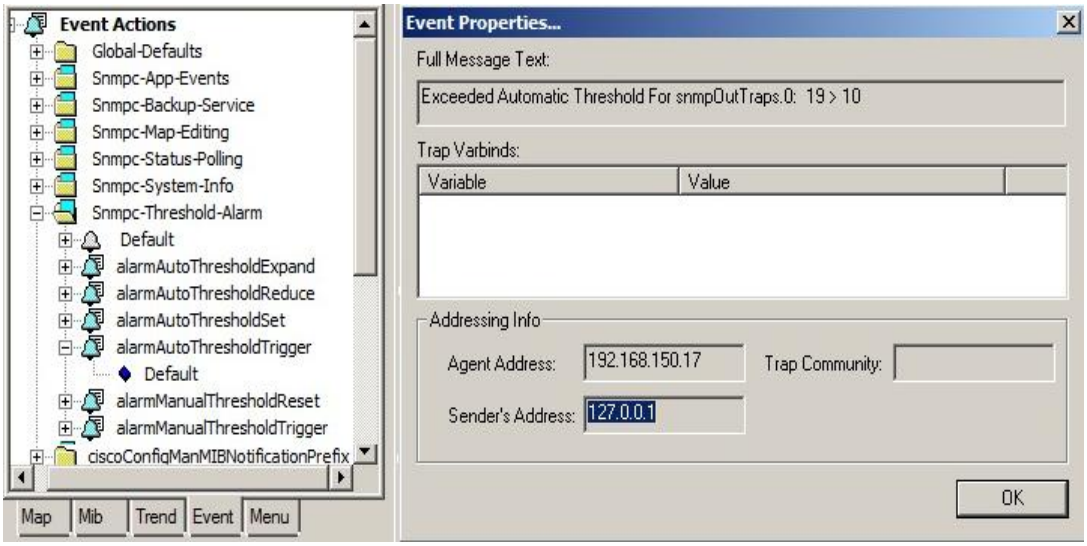
4.3.2 Nastavení automatických prahů a jejich překročení

SNMPc také umožňuje nastavení dlouhodobých statistik tak, aby byly automaticky vyhodnocovány, a v případě překročení předem definovaných hodnot se provede další akce. Tato funkce se nazývá *Trend report Treshold Alarm*. Existují dva základní druhy této funkce. První z nich je úplné automatizování všech nastavení. Systém sám vytvoří podle dlouhodobého chování systému určitou hranici, jejíž překročení si hlídá. Hlavní nastavení se provádí v záložce *Config/Trend Reports*. V okně *Trend Report Global Settings* v záložce *Automatic Alarms* je možné zapnout tyto alarmy a určit, po jak dlouhé době je možné získat alarm ze stejné proměnné instance. Instancí se myslí například množství přijatých paketů na konkrétním portu, konkrétního přepínače. Tuto hodnotu *Limit Alarm For* je přijatelnější nastavit na delší dobu. Jinak se může stát, že při dlouhodobějším překročení budeme zahlceni alarmy. Při prvním nastavení je nutné vytvořit takzvanou Baseline, což je počáteční hodnota, ze které bude systém vycházet. To zajistí funkce *Base Creation/Learning Period*. Dále je možné nastavit automatickou změnu Baseline Creation. V důsledku toho, po soustavném přijímání alarmů z jedné instance, se nastaví baseline tak, aby odpovídala novým hodnotám. Na druhou stranu pokud se nevyskytne během jedno týdne žádný alarm, tak se baseline automaticky sníží. Jak je vidět na Obrázek 27, v logu jsou zaznamenány překročené hodnoty. Dozvíme se zde, že jde o alarm s danou prioritou, čas vyvolání alarmu, prvek, který alarm zahlásí a výpis konkrétních hodnot. Překročení automatické hladiny například na vstupu portu 1 v bit/s.



Obrázek 27 : Automatický alarm, log

Tyto automatické alarmy lze upravovat, nebo vytvářet nové v záložce *Event/Event Actions*, zde si vybereme patřičnou složku *Snmpc-Treshold-Alarm*. Oznamování překročení prahových hodnot je možné provádět alarmAutoTresholdTrigger, dále také rozšíření prahových úrovní v alarmAutoTresholdExpand a podobně. Obrázek 28 znázorňuje výběr patřičné události, v pravé části vidíme, že byl vyvolán po překročení deseti odeslaných trap zprávách ze zařízení. Zařízení odsílající trapy má IP adresu lokální 192.168.150.17, což je adresa portu směrovače E. Adresa zařízení generujícího alarm je 127.0.0.1, což je adresa lokální smyčky počítače. Alarm tedy vygeneroval sám SNMPc manažer.



Obrázek 28: Automatický alarm, event

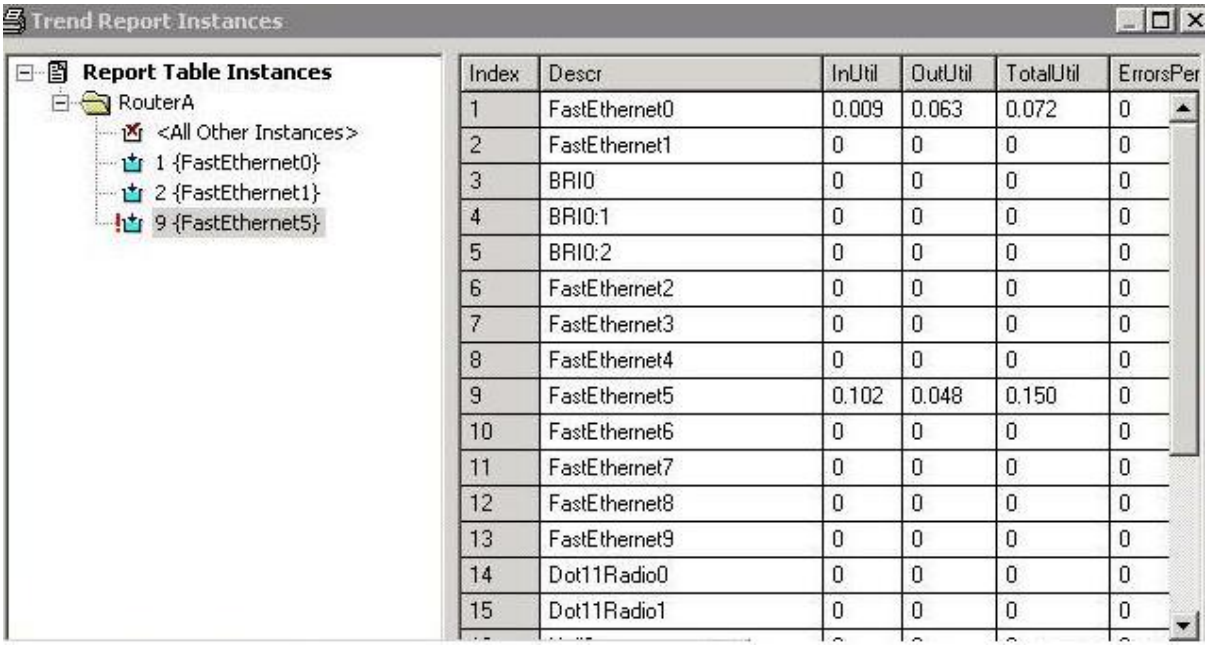
4.3.3 Nastavení manuálních prahů a jejich překročení

Pro administrátora je také důležité moci vytvořit takové prahy a události, které budou neměnné a vždy zareagují na ten samý podnět. Může se jednat například o dostatek volné datové kapacity na FTP serveru, nebo naopak klesnutí požadované kvality služeb pod určitou mez. K tomuto účelu slouží manuální prahy. Tyto eventy je možné editovat ve stejné záložce jako eventy na automatické alarmy, avšak vytvářejí se poněkud odlišně.

Jelikož se jedná o ruční konfiguraci, bude muset administrátor takovéto prahy nejprve sám vytvořit. Je také možné vytvořit prahy pro veliké množství různých hodnot, prakticky lze vytvořit práh pro jakoukoliv proměnnou hodnotu zaznamenanou v MIB tabulce zařízení.

Pro dlouhodobé monitorování překračování prahu je nutné vytvořit Trend Report Long Term Statiku (viz 4.3.1). Na příkladu v experimentální síti budeme simulovat prostředí, ve kterém chce administrátor sítě zjistit, kdy vytížení jedné konkrétní linky přesáhne určitou mez. Na síti 192.168.120.0 je umístěn server poskytující libovolné služby. Našim cílem bude vytvořit takový filtr událostí, aby nám bylo ohlášeno překročení dvaceti procentní kapacity linky směrem k serveru i od serveru a čtyřiceti procentní překročení celkové kapacity. Tento provoz budeme sledovat na směrovači A na portu Fa 5. Dále na portech Fa 1 a 2, což jsou

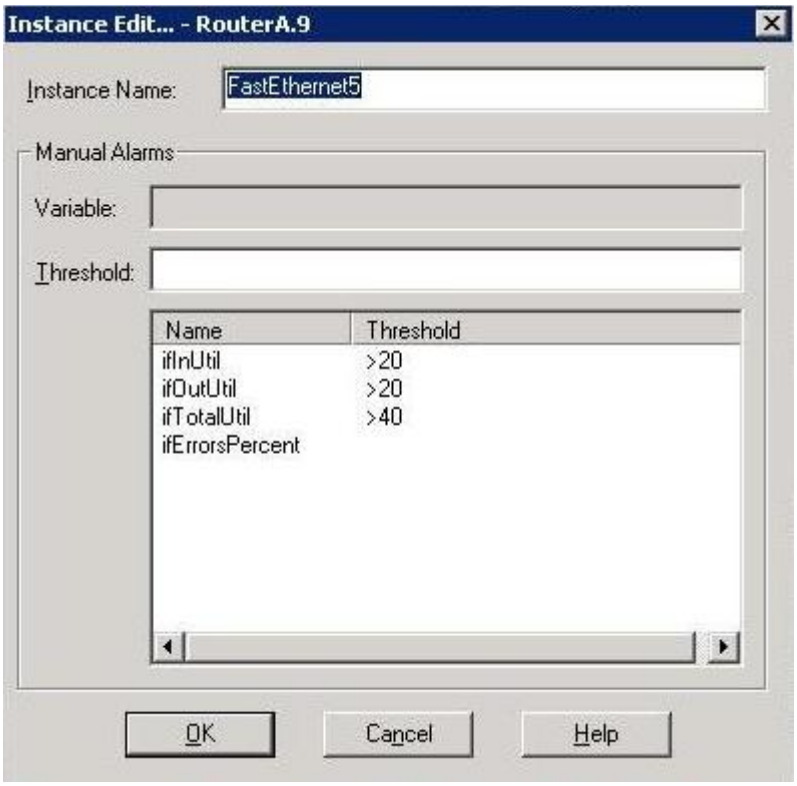
linky spojující tento směrovač s dalšími směrovači B a E (viz Obrázek 18), abychom mohli určit, kam budou data zasílána. Označíme RouterA a vytvoříme *Trend Report*. V okně *Insert Trend Report* máme na výběr několik předem definovaných možností. Například Interface Usage(BPS) – konkrétní hodnoty přenosu v bitech za sekundu, Server CPU Stats, Server Disk Stats – pro monitorování servru, Cisco System Stats – monitoruje přímo zařízení od Cisca (vytížení CPU a kapacitu paměti). Pokud nám nevyhovuje žádná z nabízejících, lze pomocí tlačítka >> přejít do prohlížeče MIB stromu a vybrat si konkrétní tabulku. Nás zajímá tabulka Interface Utilization (%). Vyplníme jméno, můžeme zde přidat nebo odebrat zařízení a upravit pollovací interval. Tyto přednastavené tabulky obsahují spoustu informací, které nám mohou nakonec při prohlížení nasbíraných dat překážet. Provedeme proto detailnější editaci tlačítkem *Instances…*. V levé části tabulky máme konečný přehled instancí, které budeme monitorovat. Implicitně je nastaveno *All Other Instances*, což jsou všechny. Označíme a zaškrtneme *Exclude*, čímž všechny zakážeme. V levé části v MIB tabulce označíme pouze řádky instance 1,2 a 9, což jsou sledované Fa porty 0,1 a 5 (viz Obrázek 29) a pomocí << je převedeme na levou stranu.



Index	Descr	InUtil	OutUtil	TotalUtil	ErrorsPer
1	FastEthernet0	0.009	0.063	0.072	0
2	FastEthernet1	0	0	0	0
3	BRI0	0	0	0	0
4	BRI0:1	0	0	0	0
5	BRI0:2	0	0	0	0
6	FastEthernet2	0	0	0	0
7	FastEthernet3	0	0	0	0
8	FastEthernet4	0	0	0	0
9	FastEthernet5	0.102	0.048	0.150	0
10	FastEthernet6	0	0	0	0
11	FastEthernet7	0	0	0	0
12	FastEthernet8	0	0	0	0
13	FastEthernet9	0	0	0	0
14	Dot11Radio0	0	0	0	0
15	Dot11Radio1	0	0	0	0

Obrázek 29: Trend Report Instance

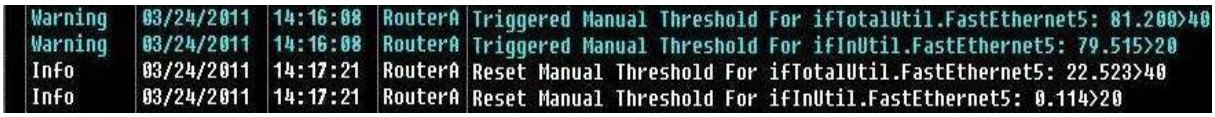
Nyní provedeme nastavení prahu instance 9. Označíme a tlačítkem *Edit* otevřeme tabulku *Instance Edit*. Jak je zobrazeno na Obrázek 30 zde můžeme nastavit požadované hodnoty, pomocí operandů =, !=, <, >, <=, >= . Uvedení jednotek zde chybí, ale jelikož se jedná o využití instance v procentech, pak vyplňujeme hodnoty 0-100. Potvrdíme všechny tabulky, můžeme také určit, jak přesně budou hodnoty zaznamenány v záložkách Export Destination. Je možné hodnoty přímo tisknout, či importovat do HTML formátu. Zvolíme denní režim a potvrdíme. V záložce Page Layout je možné specifikovat, jak bude vypadat výsledný graf, zda bude zobrazovat jednu instanci se všemi proměnnými, nebo jednu proměnnou na všech instancích.



Obrázek 30: Treshold – hodnoty a operandy

Treshold je nyní nastaven, jeho test provedeme zatížením spoje. Ze serveru 192.168.120.10 bylo streamováno video do sítě 192.168.123.0 (podsíť D) tak, aby vytížilo maximálně kapacitu spoje. Zároveň byl přenášen soubor pomalou rychlostí do sítě (podsíť C) 192.168.122.0. Administrátor se o takovéto události nejsnáze dozví z logu událostí, kde se nám zaznamená překročení prahu, jak je možné vidět na Obrázek 31. Musí si však přepnout na historii - v logu current se zobrazují pouze právě probíhající události. To znamená, že pouze v okamžiku, kdy je právě překračován práh. V momentu, kdy se hodnoty vrátí do normálu, což je čas 14:17:21, upozornění z logu current zmizí.

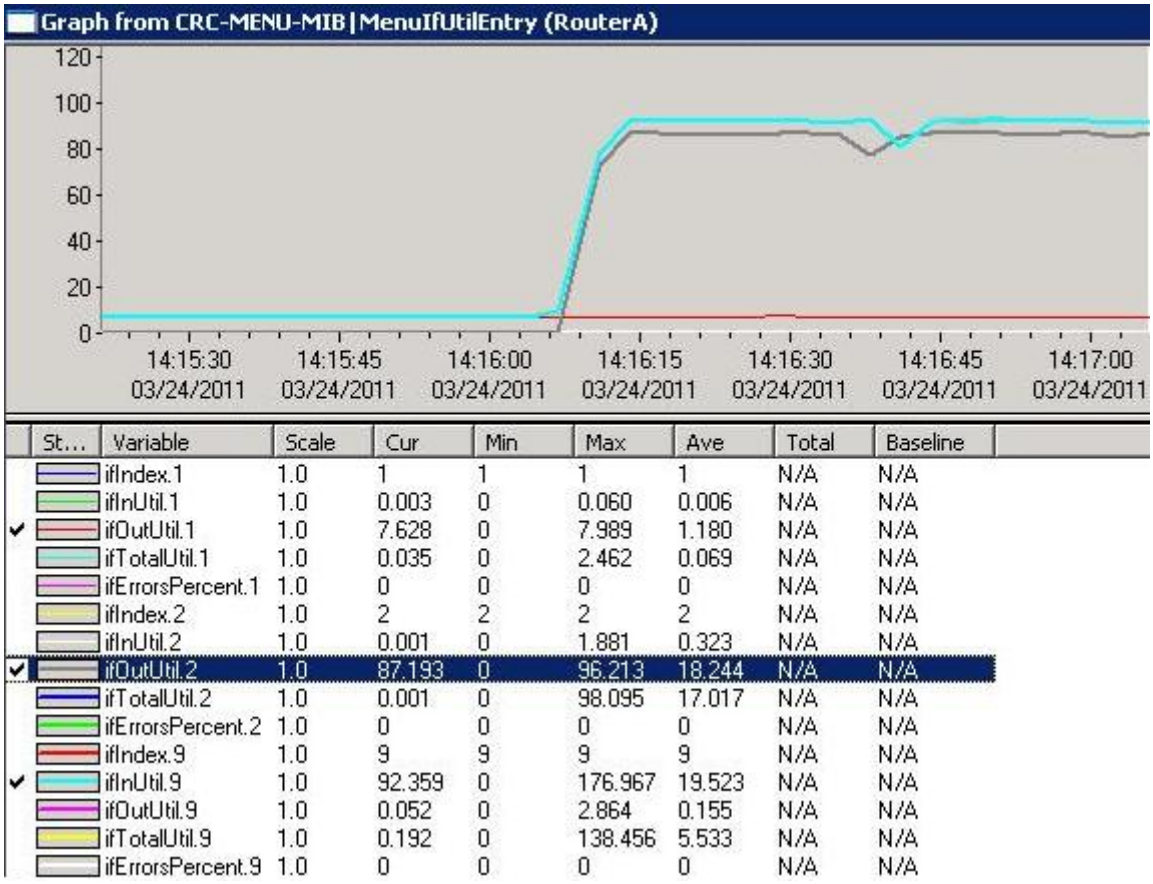
Zde se dozvíme, že byl překročen manuální Treshold na vstupu portu Fa5 směrovače A a celková kapacita na téže portu.



Obrázek 31: Treshold – log

Pro detailnější přehled o situaci si prohlédneme zaznamenaný Trend Report. V záložce Trend si pravým tlačítkem myši otevřeme volbou View Report záznam, který jsme vytvořili. V kalendáři vybereme den shodný se dnem v historii. Volbou Single Merged Graph zobrazíme všechny proměnné i instance. Jelikož jsme provedli výběr pouze pro nás důležitých instancí, bude graf přehledný. Další volby jsou zobrazení samostatných instancí, nebo proměnných, jak již bylo uvedeno výše. Provedeme zoom na lokální časovou osu v období alarmu, vidíme zde průběh vytížení (viz Obrázek 32). Po celou dobu byl zatížen port Fa0 v odchozím směru (instance ifOutUtil.1 - červená) zhruba z 8 %. Stejnou hodnotu ukazuje instance na portu Fa5 v příchozím směru (instance ifInUtil.9 - modrá). Z toho můžeme usuzovat, že provoz byl

směrován ze serveru 192.168.120.10 na směrovač B. Tento provoz by za normálních okolností nespustil alarm. V čase 14:16:08 narostlo vytížení instance ifInUtil.9 téměř na 95 %. Paralelně s tím je zaznamenáno vytížení portu Fa1 v odchozím směru (instance ifOutUtil.2 - šedá) přibližně na 87 %. Provoz byl tedy směrován ze serveru z portu Fa5 na směrovač E. Postupným zobrazováním podobných reportů na jiných směrovačích, by bylo možné dohledat cíl tohoto datového přenosu. Jednotky na Obrázek 32 jsou uvedeny na ose *x* v procentech a na ose *y* v sekundách, bohužel *x* ani *y* osa je nezobrazuje. Jednotky je možné určit pouze z popisu jednotlivých záznamu v MIB tabulce.



St...	Variable	Scale	Cur	Min	Max	Ave	Total	Baseline
	ifIndex.1	1.0	1	1	1	1	N/A	N/A
	ifInUtil.1	1.0	0.003	0	0.060	0.006	N/A	N/A
✔	ifOutUtil.1	1.0	7.628	0	7.989	1.180	N/A	N/A
	ifTotalUtil.1	1.0	0.035	0	2.462	0.069	N/A	N/A
	ifErrorsPercent.1	1.0	0	0	0	0	N/A	N/A
	ifIndex.2	1.0	2	2	2	2	N/A	N/A
	ifInUtil.2	1.0	0.001	0	1.881	0.323	N/A	N/A
✔	ifOutUtil.2	1.0	87.193	0	96.213	18.244	N/A	N/A
	ifTotalUtil.2	1.0	0.001	0	98.095	17.017	N/A	N/A
	ifErrorsPercent.2	1.0	0	0	0	0	N/A	N/A
	ifIndex.9	1.0	9	9	9	9	N/A	N/A
✔	ifInUtil.9	1.0	92.359	0	176.967	19.523	N/A	N/A
	ifOutUtil.9	1.0	0.052	0	2.864	0.155	N/A	N/A
	ifTotalUtil.9	1.0	0.192	0	138.456	5.533	N/A	N/A
	ifErrorsPercent.9	1.0	0	0	0	0	N/A	N/A

Obrázek 32: Trend Report a Treshold

4.4 VÝPADEK TRASY PŘI FTP PŘENOSU

Jedním ze měření provedených na experimentální síti je monitorování FTP přenosu a reakce systému na výpadek spojení. Využít můžeme nastavení systému z kapitoly 4.3.3. Ze serveru umístěným v lokální síti za směrovačem A budeme přenášet soubor s omezením rychlosti na 1 Mbyte/s. Cílem je stanice umístěná v síti LAN za směrovačem C. Logicky při nastavení všech linek na stejnou šířku pásma bude provoz směrován přes směrovač B. Obrázek 33 ukazuje graf vytvořený monitorováním provozem na portech směrovače A. Pro lepší přehlednost byl vytvořen jako záznam v reálném čase, jak je možné si všimnout na časové ose. Minimální rozmezí získáváním dat z dlouhodobých statistik je 1 minuta. Tímto způsobem je možné získávat data každou sekundu.

Pro tyto účely slouží v horní části konzole tlačítko *Start Graph*. Zvolením předem definované tabulky z rolovacího okna vlevo zahájíme měření. Opět ale graf bude zobrazovat nepřehledně všechny instance. Proto vybereme z MIB prohlížeče příslušnou tabulku a přes

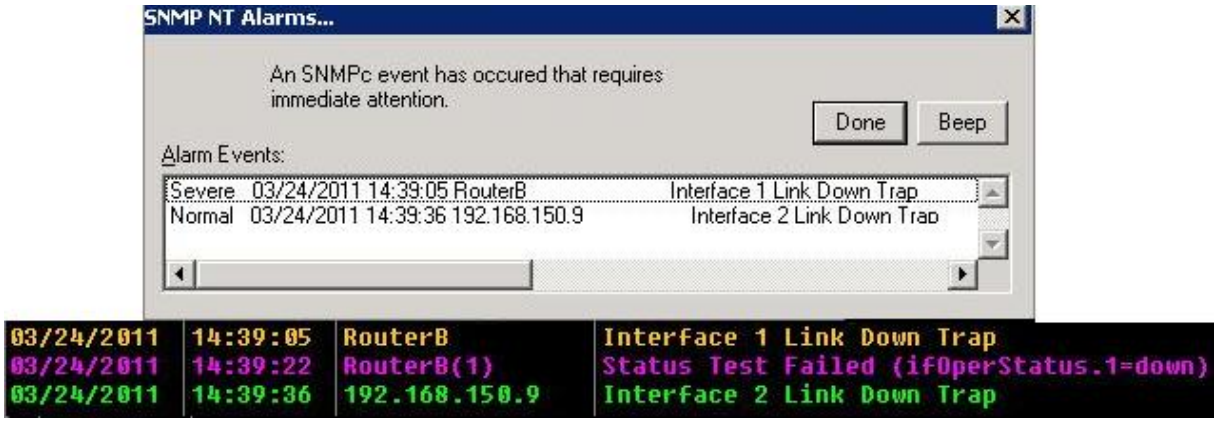
ctrl označíme pouze žádané instance. V oknu s otevřenou MIB tabulkou nahoře zvolíme pollovací interval a stiskneme *Graph*.

Samotné měření probíhalo dvě minuty. Průběh celého měření je možné odečíst z grafu na Obrázek 33. Datový tok probíhá přes port Fa5 (ifInBPS.9 - zelená) a odchází na port Fa0 (ifOutBPS.1 - červená) ke směrovači B. V čase 14:39:03 bylo přerušeno spojení mezi směrovačem B a C. Datový tok byl přesměrován protokolem OSPF delší záložní cestou přes směrovače D a E. Konkrétně na port Fa1 (ifOutBPS.2 - bílá). V čase 14:39:18 je opět obnoveno spojení a zahájen přenos. Celkový čas, kdy nebylo možné přenášet soubor, je tedy cca 15 sekund. Jak je možné si všimnout, v první části přenosu je datový tok opět směrován přes port Fa1. To je způsobeno počáteční konvergencí protokolu OSPF, která trvá cca 5 sekund. Z toho je tedy možné odvodit, že zbylých 10 sekund zpoždění, při opětovném navázání spojení, je zpusobeno protokolem FTP. V ustáleném stavu již přenos probíhá normálně.

St...	Variable	Scale	Cur	Min	Max	Ave	Total	Baseline
	ifIndex.1	1.0	1	1	1	1	N/A	N/A
	ifInBPS.1	1.0	125326	0	130501	56853	N/A	N/A
✔	ifOutBPS.1	1.0	1402	0	7801831	3122631	N/A	N/A
	ifTotalBPS.1	1.0	7929158	0	7929158	3403688	N/A	N/A
	ifIndex.2	1.0	2	2	2	2	N/A	N/A
	ifInBPS.2	1.0	342.52	0	125140	22532	N/A	N/A
✔	ifOutBPS.2	1.0	7379678	0	7822948	2472291	N/A	N/A
	ifTotalBPS.2	1.0	299.213	0	7929236	1395294	N/A	N/A
	ifIndex.9	1.0	9	9	9	9	N/A	N/A
✔	ifInBPS.9	1.0	7425311	36107	11553279	5580422	N/A	N/A
	ifOutBPS.9	1.0	156608	24828	171075	102131	N/A	N/A
	ifTotalBPS.9	1.0	7584610	60936	8235939	4364233	N/A	N/A

Obrázek 33: Přesměrování FTP přenosu

Obrázek 34 znázorňuje příjem zpráv trap ze zařízení, detekující výpadek spoje a záznam v logu. Je to jediný způsob administrátora jak zjistit tak krátký výpadek spoje. Není možné, aby sledoval provoz v reálném čase na všech zařízeních a spojích.



Obrázek 34: Přesměrování FTP přenosu - Trap

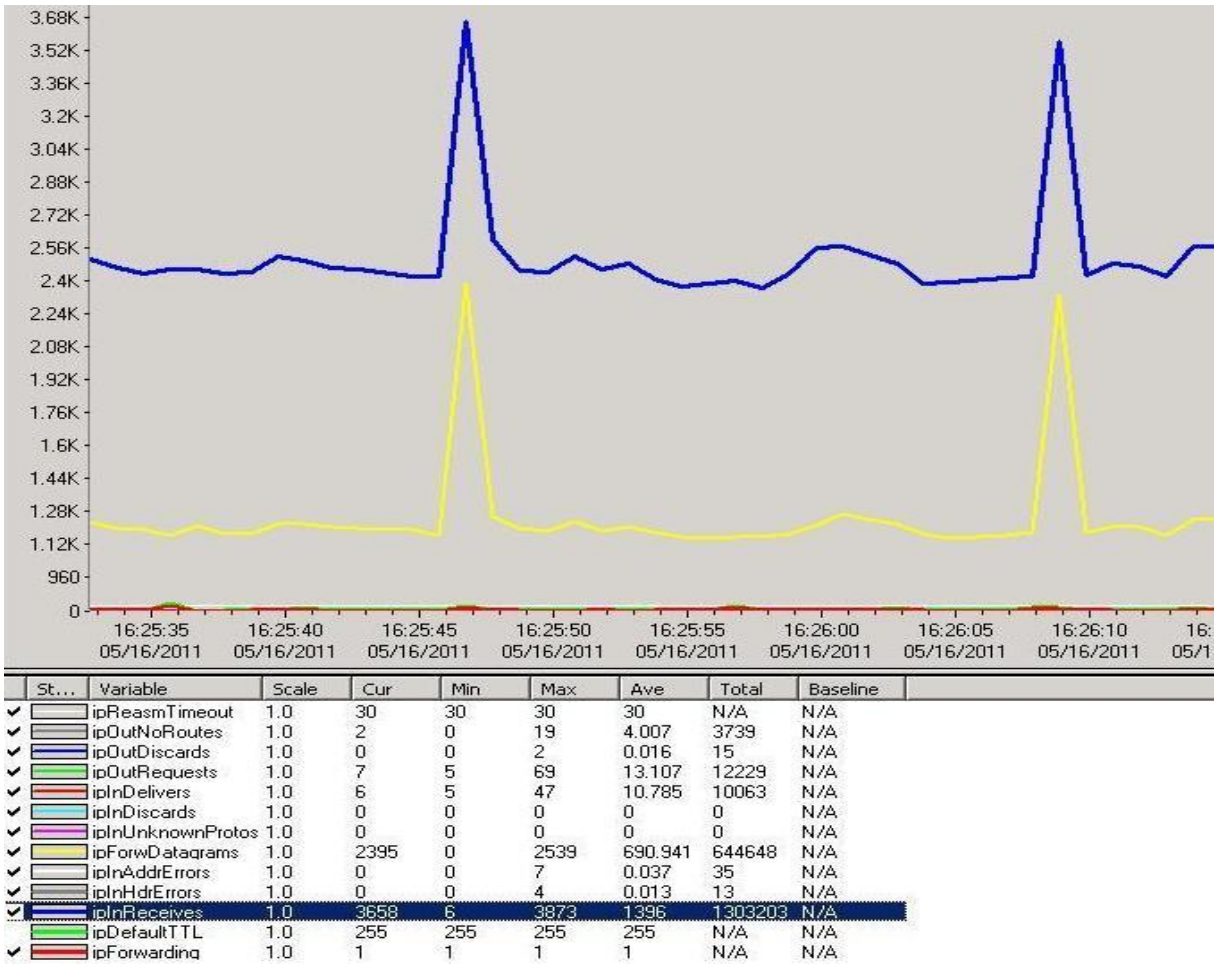
4.5 MĚŘENÍ PARAMETRŮ QOS

V experimentální síti proběhlo také měření vlivu zatížení směrovače na hovor, mezi dvěma videotelefony. Nejdříve byly na směrovači uplatněny mechanismy LLQ (Low Latency Queueing) a CBWFQ (Clase-Based Weighted Fair Queueing), které jsou blíže popsány v příloze strana 88. Jedná se o politiky rozdělování provozu pomocí tříd a vytváření prioritních front.

V rámci měření byla snížena šířka pásma spoje experimentální sítě (viz Obrázek 18) mezi směrovači A a E, zároveň rozpojen spoj kruhové topologie sítě mezi A a B. Provoz byl tedy směrován přes směrovač A a následně přes směrovač E do další části sítě.

Jako první byl realizován provoz mezi dvěma videotelefony cca v čase 16:17:00, které se nalézaly v podsítích B a C. Videotelefony komunikují přes ústřednu umístěnou na serveru s IP adresou 192.168.120.10. To znamená, že provoz ze sítě B jde do ústředny na serveru a zpět přes směrovač A do podsítě C (také v opačném směru). Videotelefony generovaly provoz o velikosti 207 kbit/s v jednom směru (celkem tedy 515 kb/s). Obrázek 35 znázorňuje okamžitý graf přenosu, vytvořený pomocí MIB tabulky *ipGroup* na směrovači A. Intance *ipInRecieves* (v grafu modrá) je množství paketů za sekundu, které byly přijato na všech portech směrovače. Žlutá barva znázorňuje instanci *ipForwDatagram*, což je množství paketů, které nebyl předány do jejich cílové destinace (byly odeslány na další směrovač). Jejich počet je tedy poloviční oproti všem přijatým paketům, jelikož polovina jich dosáhla cíle na serveru na portu Fa5 a nebyla předávána dál.

Další etapou měření bylo zvýšení provozu streamem videa ze serveru do sítě C. Video bylo streamováno rychlostí 3 Mbit/s od cca 16:33:00 hodin. Na Obrázek 36 je možné vidět, v první části cca do doby 16:38:45, graf aktivity na portech směrovače v Mbit/s v této etapě. Červená křivka (instance ifOutBPS.17) je výstupní tok na portu směrem k serveru. Jedná se 512 kbitový tok videokonference a další data (jako např. spuštěná vzdálená SNMP konzola). Zelená křivka (instance ifInBPS.17) zobrazuje data proudící v opačném směru, jedná se data video konference 2 x 512 kbit/s, data streamovaného videa 3Mbit/s a ostatní data (např. vzdálená plocha na server, či SNMP dotazy). Šedá křivka (instance ifTotalBPS.2) je pak výsledný datový tok, sčítající vstupní a výstupní data na portu Fa1 (port ke směrovači E), koresponduje s modrou křivkou (instance ifTotalBPS.17).

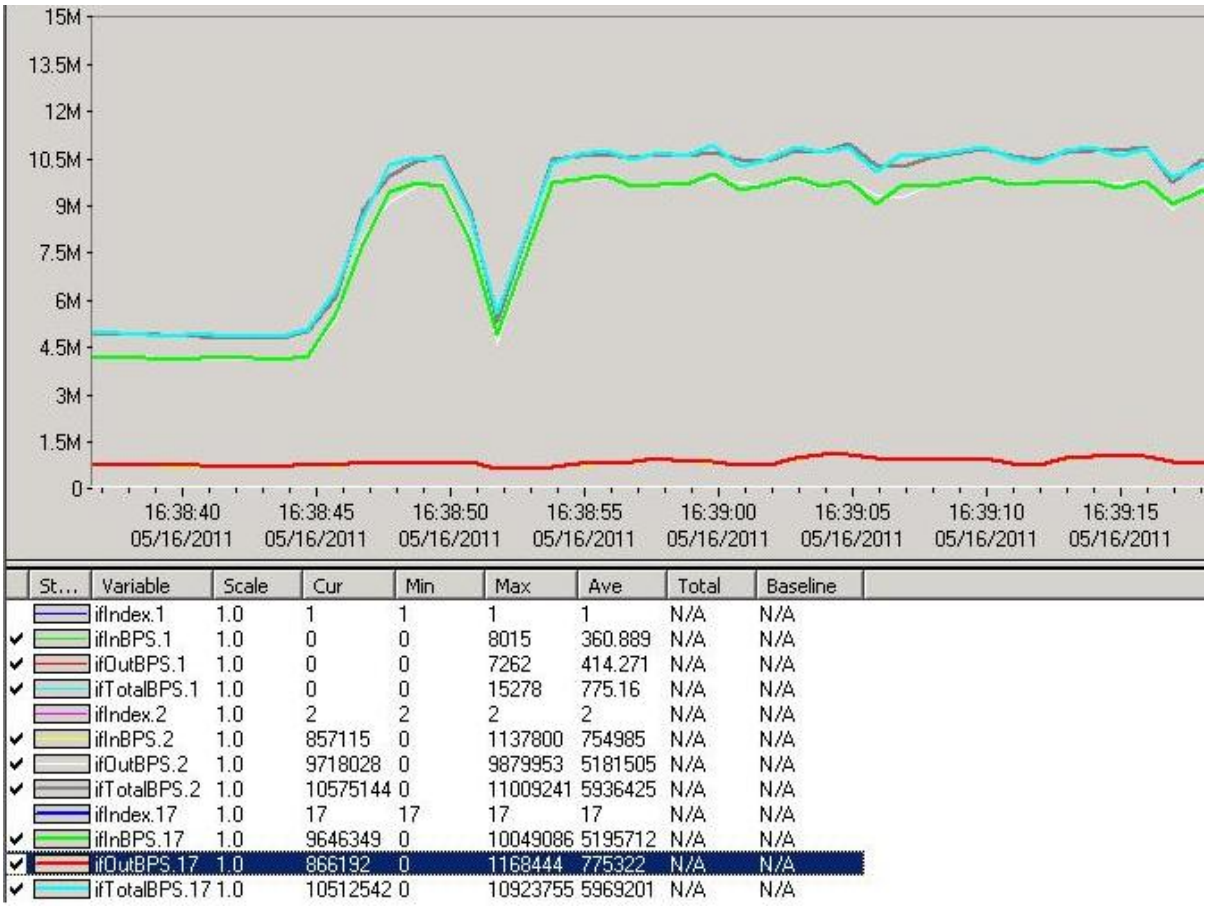


St...	Variable	Scale	Cur	Min	Max	Ave	Total	Baseline
✔	ipReasmTimeout	1.0	30	30	30	30	N/A	N/A
✔	ipOutNoRoutes	1.0	2	0	19	4.007	3739	N/A
✔	ipOutDiscards	1.0	0	0	2	0.016	15	N/A
✔	ipOutRequests	1.0	7	5	69	13.107	12229	N/A
✔	ipInDelivers	1.0	6	5	47	10.785	10063	N/A
✔	ipInDiscards	1.0	0	0	0	0	0	N/A
✔	ipInUnknownProtos	1.0	0	0	0	0	0	N/A
✔	ipForwDatagrams	1.0	2395	0	2539	690.941	644648	N/A
✔	ipInAddrErrors	1.0	0	0	7	0.037	35	N/A
✔	ipInHdrErrors	1.0	0	0	4	0.013	13	N/A
✔	ipInReceives	1.0	3658	6	3873	1396	1303203	N/A
	ipDefaultTTL	1.0	255	255	255	255	N/A	N/A
✔	ipForwarding	1.0	1	1	1	1	N/A	N/A

Obrázek 35: Měření 1 – ipGroup.0

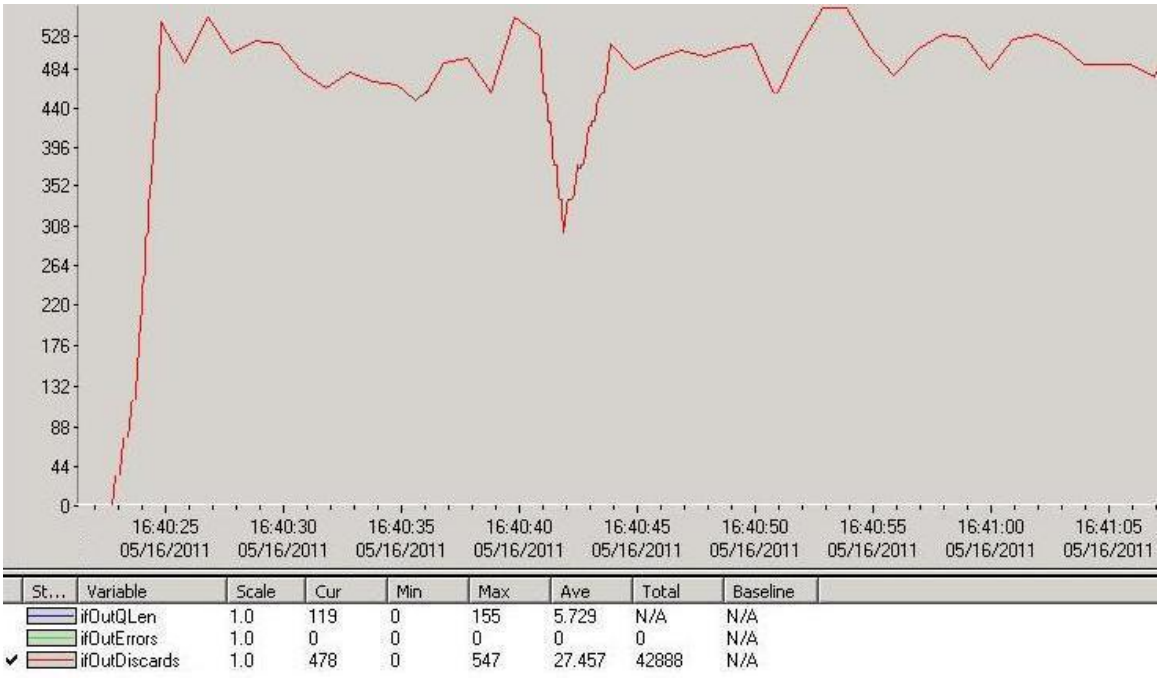
Třetí etapou měření bylo zatížení spoje přenosem FTP dat ze serveru do podsítě B v čase cca 16:38:45. Na Obrázek 36 je tento nárůst přenosu dat viditelně zaznamenán. Jak je možné vidět na Obrázek 37, který zobrazuje počet zahozených paketů, v čase do 16:40:23 stále nedochází k téměř žádnému zahazování. Je to způsobeno implementací mechanismu LLQ a CBWFQ, kdy je každé třídě provozu přiřazena jiná šířka pásma viz příloha strana 85 (Obrázek 64). V našem případě byla na směrovači A využita politika OUT_Policy_4 – CS_upřednostněný (viz Obrázek 66), kde pro videokonferenční hovor je vyhrazeno 10 % z celkových 10 Mbit, čili 1 Mbit. Stejně tak je v této politice vyhrazeno pro stream videa 15 % to je 1,5 Mbit šířka pásma. Původní rychlost streamu 3 Mbit/s se tedy musela snížit na 1,5 Mbit/s v důsledku zahlcení FTP přenosem, ten tvoří zbytek datového toku. Rychlost FTP je regulována, proto nedochází v tomto čase stále k žádnému zahazování.

Poslední etapa měření zahrnuje spuštění neznámého provozu. To bylo provedeno streamem videa na portu, který nebyl v politice na směrovači A definován. V čase 16:40:23 začíná směrovač zahazovat na výstupním portu – Obrázek 37 (červená – instance ifOutDiscards), také je způsobeno snížení přenosové rychlosti FTP na 10 % z celkové přenosové rychlosti spoje, což je 1 Mbit/s. Pakety, které jsou zahazovány, jsou složeny převážně z paketů neznámého provozu a také z malé části z FTP přenosu. Všechny tyto změny využití šířky pásma jednotlivými provozy korespondují s Obrázek 67



St...	Variable	Scale	Cur	Min	Max	Ave	Total	Baseline
	ifIndex.1	1.0	1	1	1	1	N/A	N/A
✔	ifInBPS.1	1.0	0	0	8015	360.889	N/A	N/A
✔	ifOutBPS.1	1.0	0	0	7262	414.271	N/A	N/A
✔	ifTotalBPS.1	1.0	0	0	15278	775.16	N/A	N/A
	ifIndex.2	1.0	2	2	2	2	N/A	N/A
✔	ifInBPS.2	1.0	857115	0	1137800	754985	N/A	N/A
✔	ifOutBPS.2	1.0	9718028	0	9879953	5181505	N/A	N/A
✔	ifTotalBPS.2	1.0	10575144	0	11009241	5936425	N/A	N/A
	ifIndex.17	1.0	17	17	17	17	N/A	N/A
✔	ifInBPS.17	1.0	9646349	0	10049086	5195712	N/A	N/A
✔	ifOutBPS.17	1.0	866192	0	1168444	775322	N/A	N/A
✔	ifTotalBPS.17	1.0	10512542	0	10923755	5969201	N/A	N/A

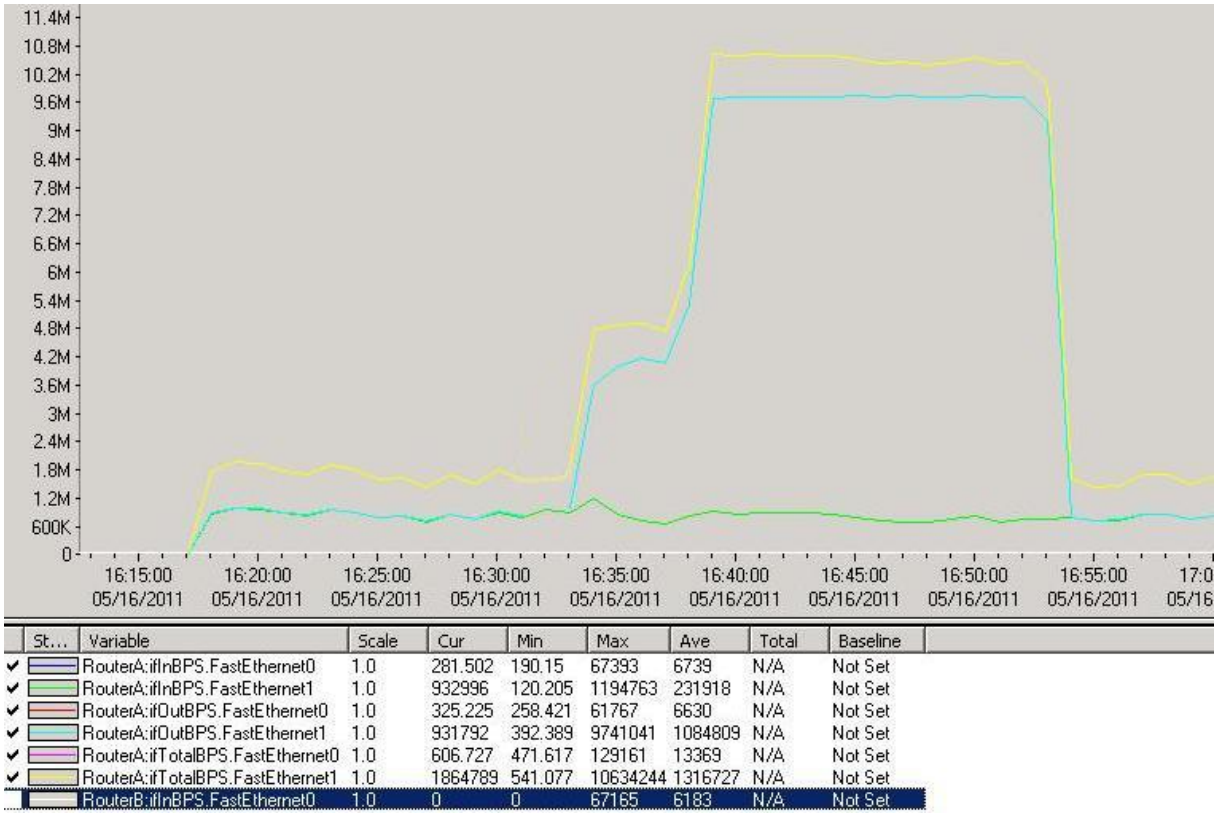
Obrázek 36: Měření 2 – ifBPSEntry



St...	Variable	Scale	Cur	Min	Max	Ave	Total	Baseline
	ifOutQLen	1.0	119	0	155	5.729	N/A	N/A
	ifOutErrors	1.0	0	0	0	0	0	N/A
✔	ifOutDiscards	1.0	478	0	547	27.457	42888	N/A

Obrázek 37: Měření 3 - zahozené pakety

Na Obrázek 38 je zobrazena jednodenní dloudobá statistika přenosu dat na směrovači A, přiblížená na sledovaný časový úsek. První etapa přenosu dat pouze videokonference probíhala cca od 16:17:00, druhá etapa streamování videa od 16:33:00. V čase od 16:38:45 proběhla třetí etapa přenosu i FTP a v čase 16:40:23 byl spuštěn neznámý provoz. Čtvrtá etapa není z grafu rozpoznatelná, jelikož program SNMPc nerozlišuje mezi různými druhy provozu a zobrazuje pouze výsledný přenos. Při aplikaci mechanismu LLQ a CBWFQ nebyl přenos videokonference ovlivněn v žádné etapě, jelikož pro něj byla plně vyhrazena třída provozu o šířce pásma 1 Mbit. Výsledný obraz i zvuk tedy byly bezezměny.



St...	Variable	Scale	Cur	Min	Max	Ave	Total	Baseline
✔	RouterA:ifInBPS.FastEthernet0	1.0	281.502	190.15	67393	6739	N/A	Not Set
✔	RouterA:ifInBPS.FastEthernet1	1.0	932996	120.205	1194763	231918	N/A	Not Set
✔	RouterA:ifOutBPS.FastEthernet0	1.0	325.225	258.421	61767	6630	N/A	Not Set
✔	RouterA:ifOutBPS.FastEthernet1	1.0	931792	392.389	9741041	1084809	N/A	Not Set
✔	RouterA:ifTotalBPS.FastEthernet0	1.0	606.727	471.617	129161	13369	N/A	Not Set
✔	RouterA:ifTotalBPS.FastEthernet1	1.0	1864789	541.077	10634244	1316727	N/A	Not Set
	RouterB:ifInBPS.FastEthernet0	1.0	0	0	67165	6183	N/A	Not Set

Obrázek 38: Měření 4 – kompletní přehled

4.6 HODNOTY SMĚROVAČE ZÍSKANÉ Z MIB PROGRAMEM SNMPC

Jak již bylo zmíněno v kapitole 2.3.3, jsou veškerá nashromážděná data ukládána do MIB databáze. Program SNMPc umožňuje tyto data přehledně zobrazovat pomocí MIB prohlížeče, který je možné vyvolat v celé řadě záložek. Základní MIB prohlížeč se nachází v levé části konzole v záložce MIB. Takto můžeme vyvolat jakákoliv data, která jsou o zařízení poskytována. Stačí zařízení (jedno, nebo více) označit a v prohlížeči MIB zvolit u patřičné tabulky pravým tlačítkem *View Table*. Pokud přesně nevíme, co která tabulka představuje, pak volbou Properties je možné zobrazit detailní informace – výpis jednotlivých proměnných, číselné i textové označení polohy tabulky ve stromu, rodiče či potomky.

MIB databáze obsahuje velkou spoustu tabulek, a proto není snadné pro nového uživatele se v ní orientovat. Program SNMPc nabízí možnost vyvolat tyto tabulky mnohem uživatelsky příjemnějším způsobem. SNMPc má velké množství předem definovaných tabulek, které je možné vyvolat vybráním patřičného zařízení a to hned ve dvou úrovních. První základní a nejpoužívanější data, lze vybrat přímo pravým tlačítkem na zařízení. V nabídce System lze získat základní informace, či stav webové služby. V nabídce Routers zase informace o

portech, jejich vytížení ARP, směrovací tabulky atd… Pokročilejší data je možné získat, přímo napojením se na zařízení. Levým dvojím kliknutím na zařízení se propojí SNMPc se zařízením, nové okno zobrazuje stav portů a nabídku v horní lište, ze které je možné vybírat patřičné MIB (u směrovače viz). Pokud je SNMP agent na zařízení správně nakonfigurován, tak lze porty zařízení deaktivovat přímo z této tabulky dvojím klikem.

Obrázek 39: HubView – směrovač A

Pro praktickou ukázku jaké MIB tabulky může zařízeni nabízet, zvolíme opět směrovač A. Z MIB tabulky směrovače je možné zjistit například systémové informace (Obrázek 40) – tabulka *SystemInfo* (OID 1.3.6.1.2.1.1). Název zařízení, ID, čas od posledního restartu, kontaktní osobu, lokaci a počet běžících služeb (viz Obrázek 40).

Descr	Cisco IOS Software, C181X
ObjectID	ciscoProducts.642
UpTime	15 days 18:12:46.86
Contact	Patala
Name	RouterA
Location	192.168.120.1
Services	78

Obrázek 40: SystemInfo

Chronologicky následují tabulkou, kterou je možné získat ze směrovače, je tabulka *ifEntry* (OID 1.3.6.1.2.1.2.2.1). Tato tabulka (Obrázek 41) obsahuje informace aplikované na jednotlivých rozhraních. V tomto přehledu jsou jednotlivá rozhraní – instance (Fa0-9, vlan, atd…) zobrazeny ve sloupci *Descr* (první sloupec je Index, nezobrazen). V dalších sloupcích jsou pak jednotlivé hodnoty:

ifIndex – unikátní hodnota, větší než 0 pro každé rozhraní,
ifDescr – textový řetězec obsahující informace o daném rozhraní,
ifType – typ rozhraní,
ifMtu – maximální velikost paketu v bytech, který může být přijat na daném rozhraní,
ifSpeed – vyměřená velikost pásma rozhraní v bit/s. Tato hodnota je zaokrouhlená,
ifPhysAddress – fyzická adresa rozhraní. Pro 802.x standard MAC adresa,
ifAdminStatus – požadovaný stav rozhraní. Hodnoty up, down, testing,
ifOperStatus – okamžitý operační stav,
ifLastChange – doba od poslední změny stavu rozhraní,
ifInOctets – celkový počet všech oktetů které byly přijaty tímto rozhraním,
ifInUcastPkts – celkový počet unicast paketů, které byly doručeny vyšší vrstvě,
ifInNUcastPkts – celkový počet předaných paketů vyšší vrstvě, adresovaných jako multicast, nebo broadcast,
ifInDiscards – celkový počet všech příchozích paketů, které byly zahozeny,
ifInErrors – celkový počet doručených paketů které byly vyhodnoceny jako chybně přijaté,

ifInUnknownProtos – počet paketů, které byli přijaty na rozhraní a nebyl rozpoznán protokol,
ifOutOctets – celkový počet oktetů odeslaných z tohoto rozhraní,
ifOutUcastPkts – celkový počet unicast paketů odeslaných z tohoto rozhraní,
ifOutNUcastPkts – počet broadcast, nebo multicast paketů odeslaných z tohoto rozhraní,
ifOutDiscards – celkový počet paktů odeslaných z tohoto rozhraní, které byly zahozeny,
ifOutErrors – počet odeslaných paketů, které nemohly být doručeny z důvodu chyby,
IfOutQlen – velikost fronty ochozích paketů.

Descr	Type	Mtu	Speed	PhysAddress	AdminStatu	InOctets	InUcastPkts	InNUcastPkts	InL	OutOctets
FastEthernet0	ethernetCsmacd	1500	100000000	00 23 33 9e e7 96	up	667461369	15575695	164875	0	3274753369
FastEthernet1	ethernetCsmacd	1500	100000000	00 23 33 9e e7 97	up	342255378	1711141	104927	0	1561472694
BRI0	lapd	1500	16000		down	0	0	0	0	0
BRI0:1	propPointToPointSerial	1500	64000		down	0	0	0	0	0
BRI0:2	propPointToPointSerial	1500	64000		down	0	0	0	0	0
FastEthernet2	ethernetCsmacd	1500	100000000	00 23 33 9e e7 98	up	0	0	0	0	0
FastEthernet3	ethernetCsmacd	1500	100000000	00 23 33 9e e7 99	up	0	0	0	0	0
FastEthernet4	ethernetCsmacd	1500	100000000	00 23 33 9e e7 9a	up	0	0	0	0	0
FastEthernet5	ethernetCsmacd	1500	100000000	00 23 33 9e e7 9b	up	172188450	35378685	342687	0	1016150894
FastEthernet6	ethernetCsmacd	1500	100000000	00 23 33 9e e7 9c	up	11977047	80888	30192	0	109590238
FastEthernet7	ethernetCsmacd	1500	100000000	00 23 33 9e e7 9d	up	0	0	0	0	0
FastEthernet8	ethernetCsmacd	1500	100000000	00 23 33 9e e7 9e	up	7037960	93751	223	0	229230736
FastEthernet9	ethernetCsmacd	1500	100000000	00 23 33 9e e7 9f	up	499412457	381990	159	0	7830353
Dot11Radio0	ieee80211	1500	54000000	00 22 55 83 07 40	down	0	0	0	0	0
Dot11Radio1	ieee80211	1500	54000000	00 22 55 83 07 30	down	0	0	0	0	0
Null0	other	1500	4294967295		up	0	0	0	0	0
Vlan1	ethernetCsmacd	1500	100000000	00 23 33 9e e7 96	up	300766711	35555742	392259	42	874807527
Loopback0	softwareLoopback	1514	4294967295		up	0	0	0	0	60
BRI0-Physical	isdns	1500	144000		down	UNK	UNK	UNK	UN	UNK
BRI0-Signaling	isdn	1500	16000		down	0	0	UNK	0	0
BRI0:1-Bearer Channel	ds0	1500	64000		down	UNK	UNK	UNK	UN	UNK
BRI0:2-Bearer Channel	ds0	1500	64000		down	UNK	UNK	UNK	UN	UNK

Obrázek 41: ifEntry

Tabulka *IpAddrEntry* (Obrázek 42) zobrazuje IPv4 adresy jednotlivých aktivních rozhraních, jejich indexy, síťovou maskou, hodnotu nejméně významného bitu v IPv4 broadcastové adrese, užívané na logickém rozhraní pro zasílání paketů. *ReasmMaxSize* je maximální velikost paketu který může být znovu sestaven po přijetí fragmentovaného paketu na tomto rozhraní.

Addr	IfIndex	NetMask	BcastAddr	ReasmMaxSize
192.168.120.1	17	255.255.255.0	1	18024
192.168.150.1	1	255.255.255.252	1	18024
192.168.150.18	2	255.255.255.252	1	18024
199.99.99.99	18	255.255.255.255	1	18024

Obrázek 42: IpAddrEntry

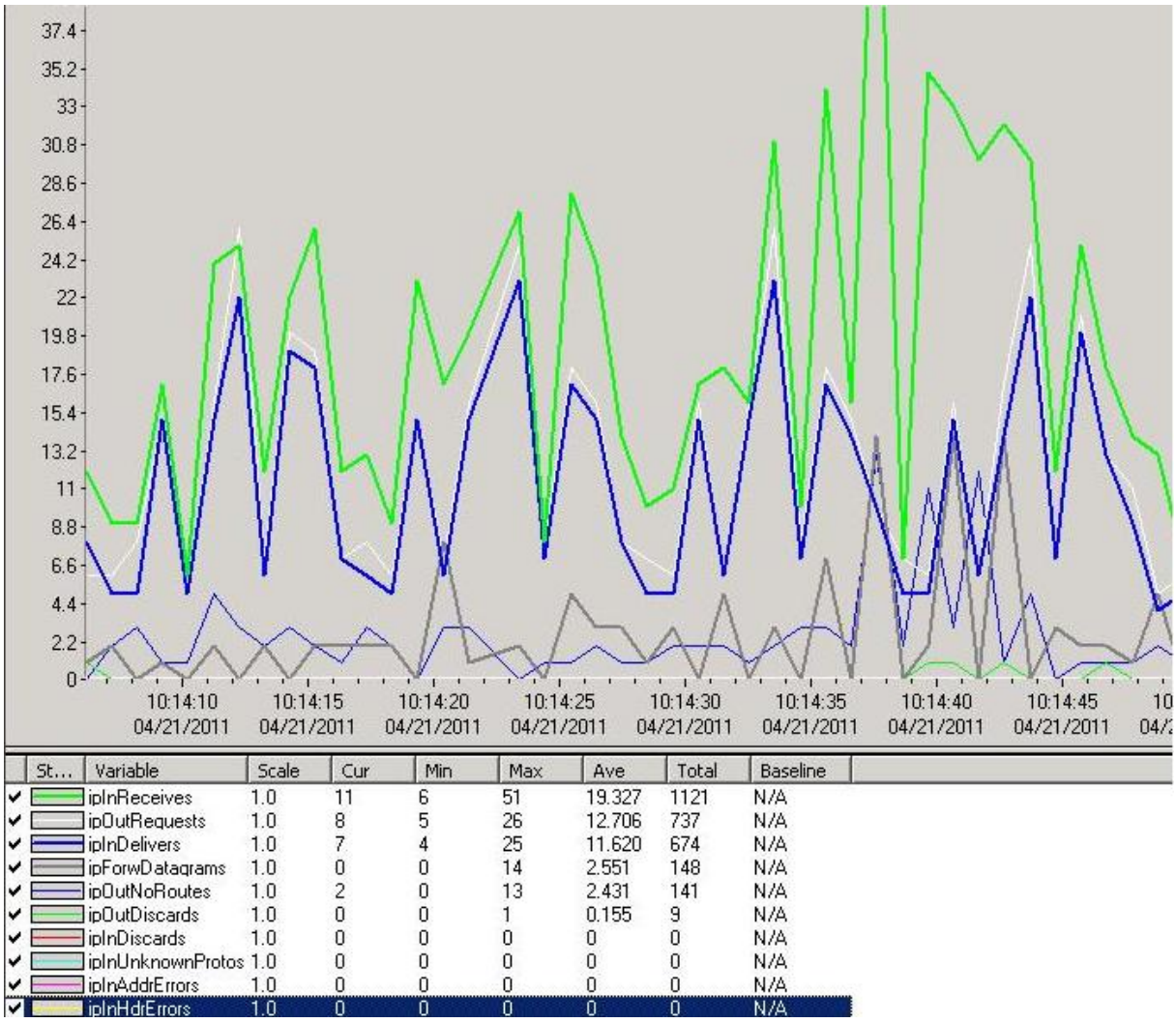
Následující tabulka *IpBasicStatsGroup* (Obrázek 43) obsahuje souhrnné informace o zařízení na síťové vrstvě. Počet přijatých a doručených paketů, počet paketů, které nebyly přímo předány do jejich cílových destinací, ale byly předány dalšímu směrovači. Položka

HdrErrors obsahuje počet všech přijatých paketů, které byly zahozeny z důvodů chybné hlavičky. AddrErrors zase počet paketu, jejichž cílová IPv4 adresa byla neplatná, například 0.0.0.0, nebo adresa z třídy E.

Receives	99911505
Delivers	495719
ForwDatagrams	47601085
OutRequests	775045
HdrErrors	7144
AddrErrors	387
UnknownProtos	0
Discards	0
OutDiscards	329230
OutNoRoutes	3877025

Obrázek 43: IpBasicStatsGroup

Z každé MIB tabulky je možno vytvořit graf, stačí pouze vybrat patřičné sloupce, či řádky a v horní části tabulky stisknout tlačítko *Graph*. Z tabulky *IpBasicStatsGroup* vybráním prvního sloupce vznikne graf jako na Obrázek 44. Je nutné mít ale na paměti, že některé hodnoty tabulek jsou prakticky statické a nemění se tak často. Kdybychom vytvářeli graf ze směrovací tabulky *IpRouteEntry* (viz Obrázek 45) zobrazily by se nám pouze vodorovné čáry, tedy kromě hodnoty *Age*.



St...	Variable	Scale	Cur	Min	Max	Ave	Total	Baseline
✔	ipInReceives	1.0	11	6	51	19.327	1121	N/A
✔	ipOutRequests	1.0	8	5	26	12.706	737	N/A
✔	ipInDelivers	1.0	7	4	25	11.620	674	N/A
✔	ipForwDatagrams	1.0	0	0	14	2.551	148	N/A
✔	ipOutNoRoutes	1.0	2	0	13	2.431	141	N/A
✔	ipOutDiscards	1.0	0	0	1	0.155	9	N/A
✔	ipInDiscards	1.0	0	0	0	0	0	N/A
✔	ipInUnknownProtos	1.0	0	0	0	0	0	N/A
✔	ipInAddrErrors	1.0	0	0	0	0	0	N/A
✔	ipInHdrErrors	1.0	0	0	0	0	0	N/A

Obrázek 44: IpBasicGroupStats - graf

Směrovací tabulku je možné vyvolat pomocí MIB tabulky *IpRouteEntry* (Obrázek 45). *Metric1* je primární směrovací metrika, pro tuto cestu. *Metric2* a výše jsou alternativní směrovací metriky určené směrovacím protokolem ve sloupci *Proto*. Jestliže protokol tyto metriky nevyužívá, je nastaveno -1. Za zmínku stojí ještě hodnota *NextHop*, což je adresa rozhraní na následujícím prvku – směrovači, přes který vede cesta k dané síti. Další hodnoty v tabulce je možné odvodit z názvu.

Dest	IfIndex	Metric1	Metric2	Metric3	Metric4	NextHop	Type	Proto	Age	Mask	Metric5	Info
192.168.120.0	17	0	-1	-1	-1	192.168.120.1	direct	local	0	255.255.255.0	-1	0.0
192.168.121.0	1	2	-1	-1	-1	192.168.150.2	indirect	ospf	778412	255.255.255.0	-1	0.0
192.168.122.0	2	3	-1	-1	-1	192.168.150.17	indirect	ospf	176911	255.255.255.0	-1	0.0
192.168.124.0	2	2	-1	-1	-1	192.168.150.17	indirect	ospf	762627	255.255.255.0	-1	0.0
192.168.150.0	1	0	-1	-1	-1	192.168.150.1	direct	local	0	255.255.255.252	-1	0.0
192.168.150.4	1	2	-1	-1	-1	192.168.150.2	indirect	ospf	767233	255.255.255.252	-1	0.0
192.168.150.12	2	2	-1	-1	-1	192.168.150.17	indirect	ospf	762627	255.255.255.252	-1	0.0
192.168.150.16	2	0	-1	-1	-1	192.168.150.18	direct	local	0	255.255.255.252	-1	0.0
199.99.99.99	18	0	-1	-1	-1	199.99.99.99	direct	local	0	255.255.255.255	-1	0.0

Obrázek 45: IpRouteEntry

Následující tabulka *IpNetToMediaEntry* (OID 1.3.6.1.2.1.4.22.1) se používá pro překlad logických adres (IP adres) na fyzické adresy (MAC adresy), tzv. arp tabulka, (viz Obrázek 46).

IfIndex	NetAddress	PhysAddress	Type
1	192.168.150.1	00 23 33 9e e7 96	static
1	192.168.150.2	00 18 b9 e3 3d c1	dynamic
2	192.168.150.17	00 1b 54 39 a1 7a	dynamic
2	192.168.150.18	00 23 33 9e e7 97	static
17	192.168.120.1	00 23 33 9e e7 96	static
17	192.168.120.10	00 30 48 d3 b0 b7	dynamic
17	192.168.120.52	00 17 31 81 1a 68	dynamic

Obrázek 46: IpNetToMediaEntry

Podrobnosti a počítadla protokolu TCP je možné zobrazit tabulkou *TcpGroup* (OID 1.3.6.1.2.1.49.2.2.1). Na Obrázek 47 můžeme vidět hodnoty, a význam některých z nich. Všechny tyto hodnot jsou platné po dobu *SysUpTime*:

RtoAlgorithm – Algoritmus užívaný pro rozhodování o znovu zaslání nepotvrzených paketů, může nabývat hodnot: other, konstant, rsre, vanj, rfc 2988,
RtoMin – minimální hodnota času v milisekundách, po kterém se zašle nový paket,
RtoMax – maximální hodnota času, využití záleží na zvolené hodnotě RtoAlgorithm,
MaxConn – maximální počet současně otevřených spojení. Hodnota -1 označuje dynamický počet,
ActiveOpens – počet právě aktivních TCP spojení. Tzn. mezi stavy SYN-SENT a CLOSED,
PassiveOpens - počet pasivních TCP spojení, která přešla přímo do stavu SYN-RCVD ze stavu LISTEN,
AttemptFails – počet spojení, která přešla do stavu CLOSED ze stavu SYN-SENT nebo SYN-RCVD a spojení, která přešla z SYN-RCVD do LISTEN
InSegs – počet přijatých segmentů, zahrnuje pouze segmenty na právě navázaných spojeních,
OutSegs – počet odeslaných segmentů, nezahrnuje znovu zaslané segmenty,
RetransSegs – počet znovuzaslaných segmentů, které byly zaslány v jednom nebo více posledních paketech.

RtoAlgorithm	vanj
RtoMin	300
RtoMax	60000
MaxConn	-1
ActiveOpens	0
PassiveOpens	2258
AttemptFails	0
EstabResets	0
CurrEstab	0
InSegs	22174
OutSegs	29124
RetransSegs	4

Obrázek 47: TcpGroup

Tabulka *UdpEntry* (OID 1.3.6.1.2.1.7.5.1) na Obrázek 48 zobrazuje informace o právě naslouchajících entitách. Adresa lokálního rozhraní připraveného naslouchat a port této naslouchající entity.

Address	Port
0.0.0.0	161
0.0.0.0	162
0.0.0.0	54456
192.168.120.1	161
192.168.120.1	54941
199.99.99.99	67
199.99.99.99	162
199.99.99.99	2887
199.99.99.99	52328

Obrázek 48: UdpEntry

Na Obrázek 49 je možné vidět tabulku *ospfIfEntry* (OID 1.3.6.1.2.1.14.7.1), která popisuje rozhraní z pohledu protokolu OSPF. Za zmínku stojí hodnoty *IfRetransInterval*, což je hodnota v sekundách mezi link-state oznámením mezi směrovači. *IfHelloInterval* je čas, po jaké době posílá směrovač paket Hello na rozhraní. *IfState* je stav rozhraní – jedná se o „určené“ rozhraní. *IfDesignatedRouter* je adresa „určeného“ směrovače.

IfIpAddress	IfType	IfAdminStat	IfRtrPriority	IfRetransInterval	IfHelloI	IfRtrDeac	IfPollInterval	IfState	IfDesignatedRouter	IfBackupDesignatedRoute
192.168.120.1	broadcast	enabled	1	5	10	40	120	designatedRouter	192.168.120.1	0.0.0.0
192.168.150.1	broadcast	enabled	1	5	10	40	120	designatedRouter	192.168.150.1	192.168.150.2
192.168.150.18	broadcast	enabled	1	5	10	40	120	designatedRouter	192.168.150.18	192.168.150.17

Obrázek 49: ospfIfEntry

Tabulka na Obrázek 50 *SnmpBasicGroup* (OID 1.3.6.1.2.1.11) zobrazuje přehled přijatých a zaslaných zpráv SNMP (viz kapitola 2.3.4).

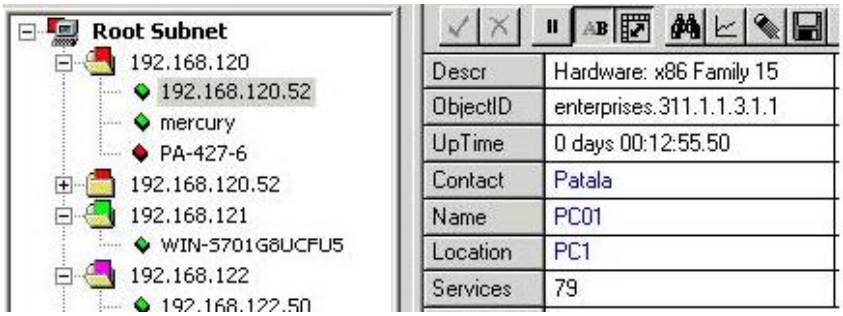
EnableAuthenTraps	enabled
InPkts	243460
OutPkts	243865
InTotalReqVars	916909
InTotalSetVars	0
InGetRequests	220176
InGetNexts	23255
InSetRequests	0
OutGetResponses	0
OutTraps	406

Obrázek 50: SnmpBasicGroup

4.7 MONITOROVÁNÍ KONCOVÉ STANICE

Po instalaci SNMP agenta na koncovou stanici (viz kapitola 3.6), je možné ji pomocí programu SNMPc monitorovat. Tento agent se také může hodit, pokud je potřeba monitorovat webový, databázový server, či diskové úložiště. Administrátorovi se nabízí také možnost získání několika zajímavých hodnot a tabulek, které monitorování jiných prvků neumožňuje.

Základní tabulkou zobrazující informace o zařízení je tabulka *SystemInfo* (OID 1.3.6.1.2.1.1) podobně jako u směrovače. Zde můžeme vidět název zařízení *Name*, kontaktní informaci *Contact*, nebo čas od restartu SNMP agenta *UpTime*.



Obrázek 51: SystemInfo

Jednoduchý výčet účtů, které jsou vedeny v operačním systému, je uveden v tabulce *svUserEntry* (viz Obrázek 52).

svUserName
fogl
Guest
Patala
Administrator
HelpAssistant
SUPPORT_388945a0

Obrázek 52: SvUserEntry

Poměrně zajímavé hodnoty jsou uloženy v MIB tabulce *HrStorageEntry* (OID 1.3.6.1.2.1.25.2.3.1) na Obrázek 53. Jedná se o výborný způsob jak mít přehled o diskových prostorách stanice. Kombinací dlouhodobé statistiky (viz 4.3.1) a nastavení prahových hodnot (viz 4.3.2) lze docílit toho, aby se při překročení určitého zaplnění disku spustil alarm. Tabulka poskytuje možnost sledovat celkovou kapacitu disku, zaplněný prostor v bytech, volný prostor v bytech a jejich procentuální hodnoty vzhledem k celkové kapacitě.

StorageIndex	StorageType	StorageDescr	BytesTotal	BytesFree	BytesUsed	PercentFree	PercentUsed
1	hrStorageRemovableDisk	A:\	0	0	0	0	0
2	hrStorageFixedDisk	C:\ Label: Serial Number	41.940G	31.919G	10.021G	76.105	23.894
3	hrStorageFixedDisk	D:\ Label:Data Serial	122.746G	115.501G	7.244G	94.097	5.902
4	hrStorageCompactDisc	Y:\ Label:VX2PVCCP_CS	620157000	0	620157000	0	100
5	hrStorageVirtualMemory	Virtual Memory	2.583G	2.170G	412615000	84.026	15.973
6	hrStorageRam	Physical Memory	1072820000	559153000	513671000	52.119	47.880

Obrázek 53: hrStorageEntry

Následující tabulka *hrSWRunEntry* na Obrázek 54 zobrazuje seznam právě běžících procesů na sledované stanici. Jsou zde zobrazeny základní informace jako je proces ID, type, status, využití procesoru a paměti, cesta a další parametry. Tyto informace je možné využívat například při kontrole zaměstnanců firmy, zda nevyužívají nepovolené programy pro komunikaci nebo sledování filmů v pracovní době. Pokud nastanou potíže například s databázovým, či poštovním serverem, administrátor je schopen pomocí této tabulky zjistit, zda komplikaci nezpůsobil nežádoucí proces, běžící na pozadí systému.

Index	Name	Type	Status	TotalCpu	PerfMem	Path	Parameters
1	System Idle Process	operatingSystem	running	28875	16		
4	System	operatingSystem	running	21.18	212		
472	mspaint.exe	application	running	2.45	29876	C:\WINDOWS\system32\	
552	alg.exe	application	running	0.07	3608	C:\WINDOWS\System32\	
636	SJphone.exe	application	running	2.78	9620	C:\Program Files\SJphone	-startup
708	smss.exe	application	running	0.07	400	\SystemRoot\System32\	
756	csrss.exe	application	running	9.32	4352	C:\WINDOWS\system32\	ObjectDirectory=\Wir
784	winlogon.exe	application	running	3.43	12624		
828	services.exe	application	running	5.71	4340	C:\WINDOWS\system32\	
840	lsass.exe	application	running	1.2	1060	C:\WINDOWS\system32\	
1016	svchost.exe	application	running	0.31	5228	C:\WINDOWS\system32\	-k DcomLaunch
1084	svchost.exe	application	running	0.78	4296	C:\WINDOWS\system32\	-k rpcss
1172	svchost.exe	application	running	9.62	27384	C:\WINDOWS\System32\	-k netsvcs
1232	svchost.exe	application	running	0.17	3536	C:\WINDOWS\system32\	-k NetworkService
1252	explorer.exe	application	running	31.06	14912	C:\WINDOWS\	
1368	dumpcap.exe	application	running	0.06	4316	C:\Program Files\Wireshark\	-i
1396	svchost.exe	application	running	0.12	3828	C:\WINDOWS\system32\	-k LocalService
1488	wscntfy.exe	application	running	0.04	2116	C:\WINDOWS\system32\	
1528	spoolsv.exe	application	running	0.14	4728	C:\WINDOWS\system32\	
1692	svchost.exe	application	running	0.07	3368	C:\WINDOWS\system32\	-k LocalService
1720	rundll32.exe	application	running	0.29	3984	C:\WINDOWS\system32\	cmicnfg.cpl,CMICtrlM
1728	ekrn.exe	application	running	27.79	61640	C:\Program	
1792	svchost.exe	application	running	0.14	4272	C:\WINDOWS\system32\	-k imgsvc
1852	egui.exe	application	running	0.76	6948	C:\Program	/hide /waitservice
1928	ctfmon.exe	application	running	0.28	3064	C:\WINDOWS\system32\	
2180	wireshark.exe	application	running	55.54	53832	C:\Program Files\Wireshark\	
2348	rundll32.exe	application	running	5.29	5540	C:\WINDOWS\system32\	
2528	snmp.exe	application	running	0.1	3312	C:\WINDOWS\System32\	
3088	mmc.exe	application	running	1.62	17228	C:\WINDOWS\system32\	
3340	snmptrap.exe	application	running	0.03	1948	C:\WINDOWS\System32\	
4036	TOTALCMD.EXE	application	running	7.17	11632	C:\totalcmd\	

Obrázek 54: hrSWRunEntry

ZÁVĚR

Diplomová práce se zaměřuje na rozbor problematiky dohledu a správy sítí pomocí protokolu SNMP. Detailním popisem protokolu se zabývá teoretický rozbor v první čtvrtině práce. Čtenář získá znalosti o historii protokolu, výhodách i nevýhodách a možnostech jeho využití. Práce uvádí rozdíly mezi třemi verzemi protokolu SNMP, které byly průběžně od roku 1989 definovány. Také uvádí detailní rozbor formátu zprávy a přenášených dat v jednotlivých polích protokolární datové jednotky. Informace byly ověřovány v síťovém protokolovém analyzátoru Wireshark. Teoretický úvod také řeší způsob uložení a získávání dat z databáze MIB.

Zbytek diplomové práce se zabývá funkcí protokolu SNMP na experimentální síti v laboratoři 427. Pro práci s tímto protokolem byl na hardwarovém serveru Merkury nainstalován manažer programu SNMPc. Ten tvoří jedno z možných efektivních řešení dohledu a zprávy počítačových sítí. V práci je detailně popsána funkce a konfigurace jednotlivých částí programu SNMPc – manažer, poller a vzdálená konzola.

K co nejefektivnější administraci experimentání sítě bylo nutné nakonfigurovat protokol SNMP i na další síťové prvky, na směrovače, přepínač a koncové stanice. Při komunikaci s těmito agenty mohl manažer SNMPc na serveru lépe získávat a vyhodnocovat dostupná data. Konfiguraci těchto síťových prvků se věnuje závěrečná část druhé kapitoly.

Zbývající nejobsáhlejší kapitola je zaměřena na simulaci reálného provozu v experimentální síti a různých situací, které mohly nastat, a konfigurací programu SNMP tak, aby co nejlépe reagoval na tyto situace. Jsou zde uvedeny simulace výpadku spoje, alarmy upozorňující správce sítě na tyto události dále zahlcení portů směrovače a vliv na kvalitu služeb (QoS). Měření QoS byly prováděny na základě spolupráce s Bc. Miroslavem Foglem a některé výroky a zjištěné poznatky odkazují na jeho diplomovou práci. Vytažek z jeho práce je uveden v příloze C.

V této kapitole je také popsán postup jak nastavit prahové limitní hodnoty na manažeru SNMPc, jak vytvořit dlouhodobé charakteristiky a záznamy měření jednotlivých parametrů síťového provozu. V závěru této kapitoly jsou uvedeny nejdůležitější hodnoty síťových prvků získané pomocí protokolu SNMP, které mohou pomoci administrátorovi pomoci optimalizovat chod sítě, a předejít tak haváriím, nebo je co nejrychleji lokalizovat a řešit. Jsou zde uvedeny například informace o portech směrovače, směrovací tabulky, informace o směrovacích protokolech, ARP tabulky, informace o UDP a TCP přenosech a podobně.Jsou zde také pospány možnosti získání informací z koncových stanic, například o diskových kapacitách, instalovaném hardwaru, operačním systému, účtech uživatelů nebo o právě běžících procesech.

K diplomové práci je také připojena příloha obsahující dvě laboratorní úlohy, které se zabývají dohledem a správou sítě pomocí protokolu SNMP. Student, který tyto úlohy absolvuje, by měl získat jak teoretické, tak praktické znalosti protokolu SNMP a jeho aplikace do komerčních systémů používajících se ke správě počítačových sítí. Na laboratorních úlohách je možné pracovat v různém pořadí a dve skupiny studentu mohou vypracovávat souběžně. Laboratorní úlohy je možné absolvovat v různém pořadí, a je možné na nich pracovat souběžně. První z nich by měla studentovi pomoci pochopit, jakým způsobem probíha komunikace mezi SNMP manažerem a agentem, vyzkouší si konfiguraci takového agenta na směrovač, a následně bude analyzovat jeho provoz sítovým analyzátorem. Druhá laboratorní úloha je zaměřena spíše na program SNMPc a seznámení se s jeho funcemi. Student v této úloze monitoruje síťové prvky a data z nich následně vyhodnocuje.

POUŽITÁ LITERATURA

[1] BIGELOW, S. J. *Mistrovství v počítačových sítích*. Computer Press, ISBN: 80-251-0178-9, ČR, 2004

[2] BURGER, A. Net-SNMP[online], 2008 [cit. 2010-11-15] Dostupné z: <http://net-snmp.sourceforge.net>

[3] *Castle Rock Dokumentace k produktu SNMPc Network Manager*. Castle Rock, 2008 Dostupné z: <http://www.castlerock.com/products/snmpc/default.php>

[4] Cisco, *SNMP object navigator* [online], 2010 [cit. 2011-5-20] Dostupné z: <http://tools.cisco.com/Support/SNMP/do/BrowseOID.do?local=en&translate=Translate&objectInput=1.3.6.1.2.1.1#oidContent>

[5] LAMMLE, Todd *Cisco Certified Network Associate*. Computer Press, ISBN 978-80-251-2359-1, ČR, 2010

[6] KASPRZAK, J. *SNMP* [online], 2007 [cit. 2010-11-04] Dostupné z: <http://www.fi.muni.cz/~kas/p090/referaty/2007-podzim/ct/snmp.html#practice>

[7] KRETCHMAN, James M.- DOSTÁLEK, Libor *Administrace a diagnostika sítí*. Computer Press, ISBN: 80-251-0345-5, 2005

[8] *NextLan Network Management* [online], 2005 [cit 2011-03-05] Dostupné z: <http://www.networkmanagement.cz/produkt.html>

[9] Novell, *Novell oznamuje WorkLoadIQ* [online], 2010 [cit. 2011-01-15] Dostupné z <http://www.novell.sk/cs/aktuality/tiskove-zpravy/novell-oznamuje-workloadiq.html >

[10]PATALA, P. *Dohledové centrum pro sítě IP*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2009. 56s. Vedoucí bakalářské práce doc. Ing. Vít Novotný, Ph.D.

[11]CASE, J.; FEDOR, M.; SCHOFFSTAL, M.; DAVIN, J. *RFC-1157 Simple Network Management Protocol (SNMP)* [online], 1990 [cit. 2010-10-15] Dostupné z: <http://www.faqs.org/rfcs/rfc1157.html>

[12]*The Internet Engineering Task Force IETF* [online], 1992 [cit 2010-11-07] Dostupné z: <http://www.ietf.org/rfc.html>

[13]KLAŠKA, L. *Svět sítí, Model Manager – Agent* [online], 2000 [cit 2010-11-07] Dostupné z: <http://www.svetsiti.cz/view.asp?rubrika=Tutorialy&clanekID=30#oper>

[14]KLAŠKA, L. *Svět sítí, Formát SNMP zpráv* [online], 2000 [cit 2010-11-07] Dostupné z: <http://www.svetsiti.cz/view.asp?rubrika=Tutorialy&temaID=23&clanekID=32>

[15]KLAŠKA, L. *Svět sítí, Síťový model správy podle ISO* [online], 2000 [cit 2010-11-08] Dostupné z: <http://www.svetsiti.cz/view.asp?rubrika=Tutorialy&temaID=23&clanekID=27>

[16]KLAŠKA, L. *Svět sítí, Smysl a přínos správy sítí* [online], 2000 [cit 2010-11-08] Dostupné z: <http://www.svetsiti.cz/view.asp?rubrika=Tutorialy&temaID=23&clanekID=24>

[17]KLAŠKA, L. *Svět sítí, Vznik a princip SNMP* [online], 2000 [cit 2010-11-10] Dostupné z: <http://www.svetsiti.cz/view.asp?rubrika=Tutorialy&temaID=23&clanekID=29>

[18]*snmpc7103* [instalační soubor] Ver. 1.0.8. Castle Rock Computing, Inc. (Saratoga, USA), 2004

[19]CASE, J.; FEDOR, M.; SCHOFFSTAL, M.; DAVIN, J *RFC-1067 Simple Network Management Protocol* (SNMP) [online], 1988 [cit 2011-3-1] Dostupné z: <http://www.faqs.org/rfcs/rfc1067.html>

[20]HOUSLEJ, R. *Authentication,confidentiality and integrity extensions to the XNS protocol SUITE* [online], 1988 [cit 2011-4-26] Dostupné z: <http://delivery.acm.org/10.1145/80000/77312/p17-

housley.pdf?key1=77312&key2=3854730031&coll=DL&dl=ACM&ip=147.229.200.111&CFID=14107997&CFTOKEN=72321496>
[21]LORIOTPRO *Network monitoring software* [online], 2011 [cit 2011-4-16] Dostupné z: <http://www.loriotpro.com/ServiceAndSupport/How_to/InstallWXPAgent_EN.php#Check>
[22]FOGL, M. *Testování podpory kvalitativních požadavků služeb v experimentální datové síti*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2011. 50 s. Vedoucí diplomové práce doc. Ing. Vít Novotný, Ph.D.
[23]MOLNÁR, K. *Uživatelem ovladatelný mechanismus diferencovaného zajištění kvality služeb*, VUT v Brně, 2007

SEZNAM ZKRATEK

ASN1	Abstract Syntax Notation One
FTP	File Transfer Protocol
ICMP	Internet Control Message Protocol
IETF	Internet Engineering Task Force
IPX	Internet Packet Exchange (Novell)
LAN	Local Area Network
MIB	Management Information Base
NMS	Network Management System
OID	Object Identifier
PDU	Protocol Data Unit
IF	Interface
Fa	FastEthernet
SNMP	Simple Network Management Protocol
ISO	International Organization for Standardization
OSPF	Open Shortest Path First
QoS	Quality of Service
LLQ	Low Latency Queueing
CBWFQ	Clase-Based Weighted Fair Queueing
TCP/IP	Transmission Control Protocol / Internet Protocol
UDP	User Datagram Protocol

SEZNAM OBRÁZKŮ

SEZNAM PŘÍLOH